AD-A040 199 NAVAL POSTGRADUATE SCHOOL MONTEREY CALIF F/G 12/2
COMPARISON OF LOG-GAMMA AND LIEBERMAN-ROSS LOWER CONFIDENCE LIM--ETC(U)
MAR 77 T R MAYNARD

UNCLASSIFIED NL

| OF |
ADA
040199

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

COMPARISON OF LOG-GAMMA AND LIEBERMAN-ROSS LOWER CONFIDENCE LIMIT PROCEDURES ON SYSTEM RELIABILITY

by

Teddy Ray Maynard

March 1977

Thesis Advisor: W. M. Woods

Approved for public release; distribution unlimited.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Comparison of Log-Gamma and Lieberman-Ross Lower Confidence Limit Procedures on System Reliability.		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis, March 1977 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Teddy Ray Maynard		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		12. REPORT DATE March 1977 13. NUMBER OF PAGES 95
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

System Reliability
Lower Confidence Limit
Log-Gamma Method
Lieberman-Ross Method
Reliability Simulation

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This thesis is an accuracy study of three empirical series system lower confidence limit procedures. Computer simulations were used to compare the accuracy of the procedures using the same set of data with 500 replications for each case. Modifications were then made to improve the accuracy of the Log-Gamma method.

→ next page

DD FORM 1 JAN 73 1473 (Page 1)

EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

UNCLASSIFIED

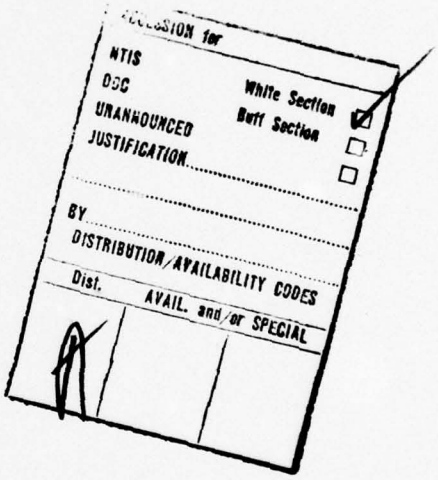
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

25145

20. (Continuation of Abstract)

The systems simulated had reliabilities ranging from .6 to .95. They were composed of four, fifteen, or forty components, and had component sample sizes of ten, twenty, or fifty.

The Lieberman-Ross method was accurate but had high variance. The Log-Gamma method was conservative with small sample sizes, but became more accurate with increased testing. Randomization of the Log-Gamma method was the most successful of the modifications attempted.



Approved for public release; distribution unlimited.

COMPARISON OF LOG-GAMMA AND LIEBERMAN-ROSS LOWER CONFIDENCE LIMIT PROCEDURES ON SYSTEM RELIABILITY

by

Teddy Ray Maynard
Lieutenant, United States Navy
B.S., Western Carolina University, 1969

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
March 1977

Author: Ted R. Maynard

Approved by: W. M. Woods
Thesis Advisor

R R Read
Second Reader

M.G. Sovereign (G K Hartman)
Chairman, Department of Operations Research

Dean of Information and Policy Sciences

ABSTRACT

This thesis is an accuracy study of three empirical series system lower confidence limit procedures. Computer simulations were used to compare the accuracy of the procedures using the same set of data with 500 replications for each case. Modifications were then made to improve the accuracy of the Log-Gamma method.

The systems simulated had reliabilities ranging from .6 to .95. They were composed of four, fifteen, or forty components, and had component sample sizes of ten, twenty, or fifty.

The Lieberman-Ross method was accurate but had high variance. The Log-Gamma method was conservative with small sample sizes, but became more accurate with increased testing. Randomization of the Log-Gamma method was the most successful of the modifications attempted.

TABLE OF CONTENTS

TABLES

TABLE OF FIGURES

ACKNOWLEDGMENT

I wish to thank Dean W. M. Woods and Professor R. R. Read for their assistance in many technical areas of this study.

I. INTRODUCTION

There exist many lower confidence limit procedures for series system reliability which are based on component data. All of them are approximate in one way or another for realistic testing situations. Two such procedures are the Lieberman-Ross (LR) procedure and the Log-Gamma (LG) procedure. The LR lower confidence limit (LCL) is based on operating time data and assumes time to failure is exponentially distributed. The LG LCL is based on attributes data and thus is non-parametric. A modification of the LG procedure, called the Continuous Log-Gamma (CLG) procedure, is based on time data and assumes an exponential failure distribution.

In this thesis the accuracy of the LR, LG, and CLG procedures are compared when data is generated from the exponential distribution for serial systems with four, fifteen, and forty components. Sample sizes are generated in a manner which is somewhat representative of data accumulated from missile flight tests.

The accuracy analysis was extended to the case where all components were assumed to have a Weibull distribution with increasing and decreasing failure rates.

In all cases, the LR and LG procedures were slightly modified from their more common forms in order to accommodate zero failures on all components.

Since the component test data was generated using the exponential assumption, the LR method enjoyed an advantage over the LG method in that continuous LCL procedures are exact while discrete LCL procedures are not. All discrete $100(1-\alpha)\%$ LCL's, $\hat{R}_{S,L(\alpha)}$, for system reliability R_s have the property that

$$P[\hat{R}_{S,L(\alpha)} \leq R_s] \geq 1 - \alpha$$

For some values of R_s, this probability is 100%. Thus discrete LCL's will usually be lower in value than a continuous LCL using the same set of data. Discrete LCL's have the advantage, however, of remaining the same regardless of the underlying distribution. Consequently, they are not susceptible to inaccuracies due to a mistaken assumption about the probability distribution of the operating component time data as are continuous procedures. That is, if $\hat{R}_{S,L(\alpha)}$ is discrete

$$P[\hat{R}_{S,L(\alpha)} \leq R_s] \geq 1 - \alpha$$

regardless of the probability distribution of the component time data, whereas if $\hat{R}^*_{S,L(\alpha)}$ is continuous

$$P[\hat{R}^*_{S,L(\alpha)} \leq R_s] = 1 - \alpha$$

provided the required assumption made about the probability distribution of the component failure data is correct. Even if the assumed continuous distribution is not correct,

however, $\hat{R}^{*}_{S,L(\alpha)}$ may still be reasonably accurate, depending upon the true component failure distribution. Of course these distributions are never known.

Finally, continuous LCL's required more refined data than discrete LCL's. Discrete LCL procedures can be used if continuous time data is available but continuous LCL procedures cannot be employed with only attributes data.

II. DESCRIPTION OF THE METHODS

A. THE LOG-GAMMA METHOD

The following development is a summary of that given in Ref. 1. The Log-Gamma (LG) method requires only attributes data, and assumes nothing about the failure distribution. It is designed for independent series systems, does not require equal component sample sizes, and is approximate. A modification of the LG method for systems composed of a mixture of series and parallel-connected components was explored by Ref. 2.

Suppose a system consists of K components in series, and that N_i copies of component i are put on test and operated until failure or completion of mission, whichever occurs first. Denote F_i as the number of components of type i that did not complete the mission. Then the maximum likelihood estimate for system reliability is

$$\hat{R}_s = \prod_{i=1}^{K} \hat{R}_i \tag{2.1}$$

where

$$\hat{R}_i = \frac{N_i - F_i}{N_i} \tag{2.2}$$

In the LG procedure the method of moments is used to fit the random variable $-\ln\hat{R}_s$ with the two-parameter gamma distribution. The gamma is then transformed into a chi

square distribution, about which probability statements are made and a lower confidence limit (LCL) obtained. That is, define

$$S = -\ln R_s = -\ln \prod_{i=1}^{K} R_i = -\sum_{i=1}^{K} \ln(1-Q_i) \tag{2.3}$$

$$\approx \sum_{i=1}^{K} (Q_i + \tfrac{1}{2}Q_i^2) \equiv \sum_{i=1}^{K} T_i \qquad \text{(See Appendix A)} \tag{2.4}$$

An unbiased estimator $\hat{T}_i$ for T_i is

$$\hat{T}_i = A_i \hat{Q}_i + B_i \frac{\hat{Q}_i^2}{2} \tag{2.5}$$

where $$A_i = \frac{2N_i - 3}{2N_i - 2} \tag{2.6}$$

$$B_i = \frac{N_i}{N_i - 1} \tag{2.7}$$

and $$\hat{Q}_i = \frac{F_i}{N_i} \qquad i = 1, K \qquad \text{(See Appendix B)}$$

That $\hat{T}_i$ is unbiased is important, because

$$\hat{S} = \sum_{i=1}^{K} \hat{T}_i \tag{2.8}$$

is used as an estimator for S, thereby accumulating any bias present in the $\hat{T}_i$. An approximate value for the variance of $\hat{S}$ is

$$\mathrm{Var}(\hat{S}) = \sum_{i=1}^{K} \mathrm{Var}(\hat{T}_i)$$

$$\approx \sum_{i=1}^{K} \frac{\hat{T}_i}{N_i} \tag{2.9}$$

Appendix C derives the actual variance of $\hat{T}_i$ and shows equation (2.9) is an excellent approximation for $\mathrm{Var}(\hat{S})$.

Next, fit $\hat{S}$ with a gamma distribution. The probability distribution of $\hat{S}$ is then given by the density function

$$f_{\hat{S}}(x;r,\theta) = \frac{1}{\Gamma(r)\,\theta^r}\, x^{r-1}\, \mathrm{Exp}(-x/\theta) \qquad \begin{matrix} x \geq 0,\ r > 0 \\ \theta > 0 \end{matrix} \tag{2.10}$$

It follows that

$$E(\hat{S}) = r\theta \tag{2.11}$$

$$\mathrm{Var}(\hat{S}) = r\theta^2 \tag{2.12}$$

Since $\hat{S}$ is unbiased,

$$E(\hat{S}) = S$$

$$= \sum_{i=1}^{K} T_i \tag{2.13}$$

Solving equations (2.9), (2.11), (2.12), and (2.13) simultaneously gives the shape parameter

$$r = \left(\sum_{i=1}^{K} T_i\right)^2 \Big/ \sum_{i=1}^{K} \frac{T_i}{N_i} \tag{2.14}$$

and the scale parameter

$$\theta = \left(\sum_{i=1}^{K} \frac{T_i}{N_i}\right) \Big/ \sum_{i=1}^{K} T_i \tag{2.15}$$

Thus, r can be estimated by

$$\hat{r} = \left(\sum_{i=1}^{K} \hat{T}_i\right)^2 \Big/ \sum_{i=1}^{K} \frac{\hat{T}_i}{N_i} \tag{2.16}$$

Since $\hat{S}$ is distributed gamma (r,θ), $\frac{2\hat{S}}{\theta}$ is distributed χ^2_{2r}. Then

$$1-\alpha = P\left(\frac{2\hat{S}}{\theta} \geq \chi^2_{1-\alpha,2r}\right) \tag{2.17}$$

$$= P\left(2r\hat{S} \geq r\theta \cdot \chi^2_{1-\alpha,2r}\right)$$

Since $\hat{S}$ is unbiased

$$E(\hat{S}) = S$$

$$= r\theta$$

$$= -\ln R_s \tag{2.18}$$

Equation (2.17) becomes

$$1-\alpha = P\left(2r\hat{S} \geq -\ln R_s \cdot \chi^2_{1-\alpha,2r}\right)$$

$$= P\left(\mathrm{Exp}\left\{\frac{-2r\hat{S}}{\chi^2_{1-\alpha,2r}}\right\} \leq R_s\right) \tag{2.19}$$

Therefore, the LG $100(1-\alpha)\%$ LCL for system reliability R_s is

$$\hat{R}_{S,L(\alpha)} = \mathrm{Exp}\left(\frac{-2\hat{r}\hat{S}}{\chi^2_{1-\alpha,2\hat{r}}}\right) \tag{2.20}$$

To preclude usage of non-integer degrees of freedom, the approximation

$$\frac{\left[2\hat{r}\right]}{\chi^2_{1-\alpha,\left[2\hat{r}\right]}} \tag{2.21}$$

is used in equation (2.20), where $[2\hat{r}]$ denotes the smallest integer greater than or equal to $2\hat{r}$. Approximation (2.21) was shown to have little effect on the LG procedure accuracy [Ref. 2, p. 13]. However, it may produce monotonicity

problems under certain circumstances (discussed in Chapter V). Thus, the 100(1-α)% LCL becomes

$$\hat{\hat{R}}_{S,L(\alpha)} = \mathrm{Exp}\left(\frac{-[2\hat{r}]\hat{s}}{\chi^2_{1-\alpha,[2\hat{r}]}}\right) \qquad (2.22)$$

and was the LG LCL initially simulated by this study (Chapter IV).

B. THE CONTINUOUS LOG-GAMMA METHOD

The Continuous Log-Gamma (CLG) method is developed in detail by Ref. 3, pages C-1 through C-20 (as the "Classical" method). It differs from the LG method only in that it requires component time data and assumes exponential failures.

The derivation of the CLG procedure is analogous to that of the LG procedure, and is outlined as follows. Define

$$Z \equiv -\ln R_s$$

$$= -\ln \prod_{i=1}^{K} R_i \qquad (2.23)$$

Using time data and assuming exponentiality, R_i can be estimated by

$$\hat{R}_i = \mathrm{Exp}\left(-\frac{F_i}{Tsum_i}\right) \qquad i=1,K$$

where $Tsum_i$ is the accumulated operating time of the ith component. Then Z is estimated by

$$\hat{Z} = \sum_{i=1}^{K} \frac{F_i}{Tsum_i} \qquad (2.24)$$

The method of moments gamma fit to $\hat{Z}$ uses

$$E(\hat{Z}) = \sum_{i=1}^{K} \frac{F_i}{Tsum_i} \tag{2.25}$$

$$\widehat{Var}(\hat{Z}) = \sum_{i=1}^{K} \left(\frac{F_i}{Tsum_i}\right) \Big/ N_i \tag{2.26}$$

$$E(\hat{Z}) = L\lambda \tag{2.27}$$

and $$Var(\hat{Z}) = L\lambda^2 \tag{2.28}$$

where L is the shape parameter and λ is the scale parameter. The resultant CLG $100(1-\alpha)\%$ LCL is

$$\hat{R}_{S,L(\alpha)} = Exp\left(\frac{-2\hat{L}\hat{Z}}{\chi^2_{1-\alpha,2\hat{L}}}\right) \tag{2.29}$$

with

$$\hat{L} = \left(\sum_{i=1}^{K} \frac{F_i}{Tsum_i}\right)^2 \Big/ \sum_{i=1}^{K} \left(\frac{F_i}{Tsum_i} \Big/ N_i\right) \tag{2.30}$$

To preclude non-integer degrees of freedom

$$\hat{\hat{R}}_{S,L(\alpha)} = Exp\left(\frac{-\left[2\hat{L}\right]\hat{Z}}{\chi^2_{1-\alpha,\left[2\hat{L}\right]}}\right) \tag{2.31}$$

where $\left[2\hat{L}\right]$ is the smallest integer greater than or equal to $2\hat{L}$.

C. THE LIEBERMAN-ROSS METHOD

1. The Original Procedure

The Lieberman-Ross (LR) method was designed as an exact lower confidence limit (LCL) procedure for independent series systems (see Ref. 4). It requires component times between successive failures from an assumed exponential failure distribution, and does not require equal component sample sizes.

The test plan may be described as follows. Suppose N_i copies of each component type i are available for testing. Place a copy of the ith component on test, and replace it when it fails. Continue testing with replacement until all available copies of one component are exhausted (this procedure need not be conducted simultaneously for all component types). For the ith component, denote the time to the first failure as T_{i1}, between the first and second failure as T_{i2}, and so forth. In general, the time between the jth and (j-1)th failure on the ith component is T_{ij}, $i = 1,\ldots,K$ $j = 1,\ldots,N_i$ where the T_{ij} are in mission units. Reference 4, p. 840, also describes the LR procedure for Type II censoring at the rth failure of R items on simultaneous test. Reference 3, p. C-32, illustrates the procedure using component times between successive failures and the observed success time of the last copy of each type tested.

The LR LCL is computed as follows. Define

$$U \equiv \min_{1 \le i \le K} \left(\sum_{j=1}^{N_i} T_{ij} \right) \tag{2.32}$$

Let I_i be the greatest index such that

$$\sum_{j=1}^{I_i} T_{ij} \le U \qquad i = 1,\ldots,K \tag{2.33}$$

Also define

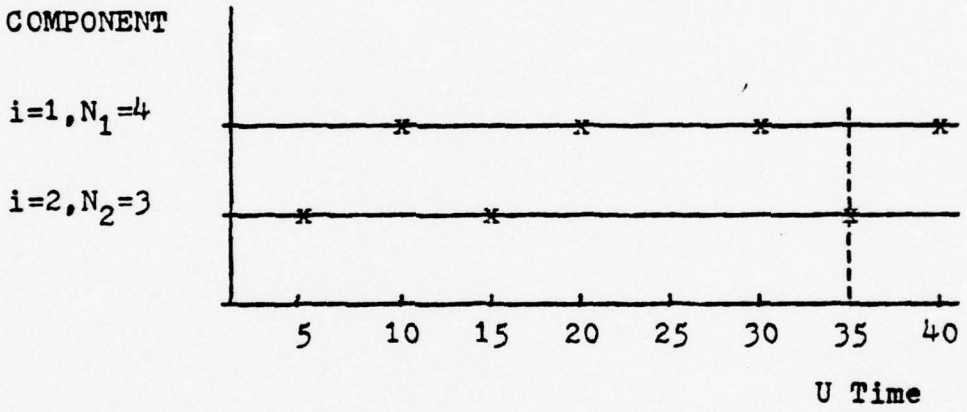
$$I \equiv \sum_{i=1}^{K} I_i \tag{2.34}$$

The variable I is the total number of failures in all components up to and including U (see Figure 1). Then the LR 100(1-α)% LCL for system reliability R_s is given by

$$\hat{R}_{S,L(\alpha)} = \text{Exp}\left(\frac{-\chi^2_{\alpha,2I}}{2U}\right) \qquad (2.35)$$

Figure 1

Depiction of Lieberman-Ross Parameters
U=35 and I=6, when K=2, N_1=4, and N_2=3



To implement the LR procedure, the component data order must be specified. If only a set of component data were available, then a different procedure would have to be used. Note that the opposite chi square tail is used from that of the LG and CLG methods.

2. <u>The Modified Lieberman-Ross Method</u>

The original procedure utilizes accumulated component times between failures. The Modified Lieberman-Ross procedure (MLR), uses instead component times to one mission unit or failure, whichever occurs first. The MLR method is thereby more readily applied than the LR method to actual

system test data, especially when testing is destructive (for example, a missile firing). The MLR method also can be used when no failures occur, whereas the LR method could not (see Appendix D).

The MLR procedure is

a. Suppose N_i copies of the ith component are available for testing. Define T_{ij} as for the LR procedure, except all times past one mission unit are truncated to one. Next, define

$$T^*_{i1} = \sum_{j=1}^{i_1} T_{ij} \qquad \text{where } i_1 \text{ is the index of the first component time less than one mission unit for the ith component} \qquad (2.36)$$

$$T^*_{i2} = \sum_{j=i_1+1}^{i_2} T_{ij} \qquad \text{where } i_2 \text{ is the index of the second component time less than one mission unit for the ith component} \qquad (2.37)$$

$$\vdots$$

$$T^*_{iF_i} = \sum_{j=i_{(F_i-1)}+1}^{i_{F_i}} T_{ij} \qquad \text{where } iF_i \text{ is the index of the last component time less than one mission unit for the ith component} \qquad (2.38)$$

Then obtain

$$T^*_i = \sum_{j=1}^{F_i} T^*_{ij} \qquad i = 1,\ldots,K \qquad (2.39)$$

b. Compute the minimum accumulated times between successive failures in any component type.

$$U = \begin{cases} \min\limits_{1\le i\le K} T^*_i & \text{if } \sum\limits_{i=1}^{K} F_i > 0 \qquad (2.40) \\ \min\limits_{1\le i\le K} N_i & \text{if } \sum\limits_{i=1}^{K} F_i = 0 \qquad \text{(see Appendix D)} \end{cases}$$

The variable I_i is now defined in terms of the T_i^*.

$$I_i \equiv \begin{cases} \text{the greatest index such that} \\ \quad \sum_{j=1}^{I_i} T_{ij}^* \leq U & \text{if } F_i > 0 \\ \quad 0 & \text{if } F_i = 0 \end{cases} \tag{2.41}$$

Then

$$I \equiv \max\left\{1, \sum_{i=1}^{K} I_i\right\} \tag{2.42}$$

c. Now the MLR $100(1-\alpha)\%$ LCL for R_s can be computed

$$\hat{R}_{S,L(\alpha)} = \text{Exp}\left\{\frac{-\chi^2_{\alpha,2I}}{2U}\right\} \tag{2.43}$$

Like the LR method, the component data order must be specified.

III. THE SIMULATION

A. GENERAL METHODOLOGY

Any lower confidence limit (LCL) procedure can be evaluated by computer simulation of its distribution as follows:

1. Suppose it is desired to evaluate a proposed $100(1-\alpha)\%$ LCL procedure, denoted $\hat{R}_{S,L(\alpha)}$, for system reliability R_s. Then the assertion is

$$P\left[\hat{R}_{S,L(\alpha)} \leq R_S\right] \geq 1 - \alpha \qquad (3.1)$$

Equality should hold if $\hat{R}_{S,L(\alpha)}$ is a continuous random variable.

2. $R_s = f(R_1, R_2, \ldots, R_k)$, where R_i is the true reliability of the ith component. For an independent series system, $R_s = \prod_{i=1}^{K} R_i$. If exponentiality is assumed, an equivalent statement is $R_s = f(\lambda_1, \lambda_2, \ldots, \lambda_k)$.

3. Assign values to the parameters α, K, and R_i (or λ_i), $i = 1,2,\ldots,K$. Simulate life testing by generating random component times-to-failure, and compute the resultant $\hat{R}_{S,L(\alpha)}$.

4. Generate the approximate distribution of $\hat{R}_{S,L(\alpha)}$ by repeating step 3 five hundred times.

5. Order the $\hat{R}_{S,L(\alpha)}$ realizations to get $\hat{R}_{S,L(\alpha)_{(1)}}, \ldots, \hat{R}_{S,L(\alpha)_{(500)}}$.

6. Find the $(1-\alpha)500$th order statistic of $\hat{R}_{S,L(\alpha)}$ and denote it $A_{1-\alpha}$. Thus $A_{1-\alpha}$ is the $(1-\alpha)$th percentile of the distribution of the LCL random variable $\hat{R}_{S,L(\alpha)}$. If in fact, $\hat{R}_{S,L(\alpha)}$ is an exact $100(1-\alpha)\%$ LCL procedure for R_s, then $A_{1-\alpha}$ should be approximately equal to R_s.

7. By repeating steps 3 through 6 for various sets of $K, N_1, N_2, \ldots, N_k$, $R_1, R_2, \ldots, R_k$ and α, and comparing the resultant $A_{1-\alpha}$ to R_s, the overall performance of the LCL procedure can be evaluated.

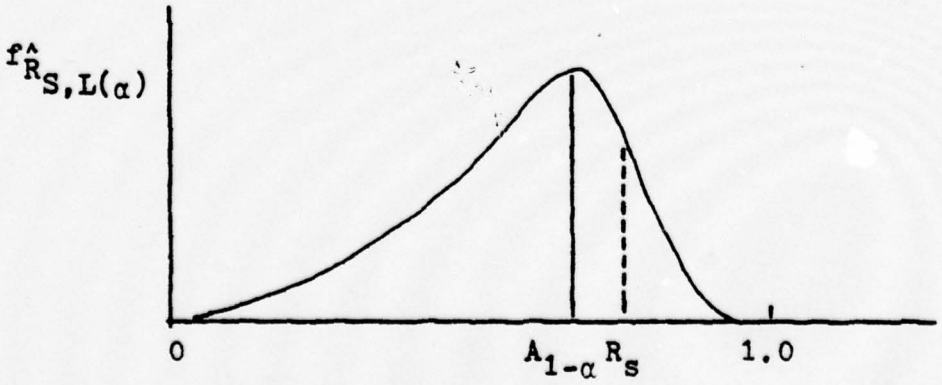
The actual confidence level given by $\hat{R}_{S,L(\alpha)}$ can be obtained by finding the order statistic of the generated distribution which matches (or closest matches) R_s. If the index of this order statistic is denoted i^*, then

$$(100\%)\frac{i^*}{500} = \text{Actual Level of Confidence of } \hat{R}_{S,L(\alpha)} \qquad (3.2)$$

Equivalently, if $A_{1-\alpha} < R_s$, the procedure is a conservative one, and vice-versa.

Figure 2 depicts a conservative outcome with $A_{1-\alpha} < R_s$. The degree to which $A_{1-\alpha}$ equals R_s reflects the validity of equation (3.1) and therefore the accuracy of $\hat{R}_{S,L(\alpha)}$.

Figure 2



Not only should an LCL procedure be accurate, but also should have small standard deviation. Even though $\hat{R}_{S,L(\alpha)}$ may be an exact LCL procedure, its practical value is diminished if it produces widely dispersed LCL's. This study used both criteria for comparison of the three methods.

B. GENERATION OF SAMPLE SIZES

The component test data are generated in a manner somewhat representative of data accumulated in missile flight tests. Suppose some number of series systems are to be used for testing. Failure of a component may fail the system, but still allow testing on components upstream of the failed one. For example, if the system is a five-stage missile, failure of the third stage may or may not allow testing on stages four and five.

Input parameter N_1, the number of mission tests on the first component, is the number of systems to be tested. A component failure is assumed to preclude testing on upstream components one-half the time. The simulation computes the number of tests on the ith component as

$$N_i = N_1 - \left[\tfrac{1}{2} \sum_{j=1}^{i-1} F_j\right] \qquad (3.3)$$

where "[ ]" denotes "the smallest integer greater than or equal to". To illustrate, let N_1=20 and assume the first component had three failures. Then the component two sample size becomes

$$N_2 = 20 - \left[\tfrac{1}{2}(3)\right] = 18 \text{ mission trials}$$

It should be understood, however, that the three procedures themselves do not require this scheme, but are valid for varying sample sizes.

C. THE SIMULATION ALGORITHM

Step one: Initilize the number of series components K, the number of systems to be tested N_1, the true reliability of the ith component R_i, i=1,2,...,K, and the number of replications desired ("Do Loop" index).

Step two: Using the probability integral transform (PIT) method [Ref. 5, p. 168], generate exponential failure times for the N_1 copies of the first component as shown below. Exponential failure data is a requirement for the CLG and LR methods. The LG method, being non-parametric, can use the same data.

a. Generate N_1 uniform (0,1) random numbers for component one. Denote these $U_{11}, U_{12}, \ldots\ldots, U_{1N_1}$

b. Let $F_{1j} = \begin{cases} 0 & \text{if a success occurred for component one on trial } j \\ 1 & \text{if component one failed on trial } j,\ j=1,2,\ldots,N_1 \end{cases}$

Appendix E shows

$$F_{1j} = \begin{cases} 0 & \text{if } 0 \le U_{1j} < \text{Exp}(-\lambda_i) \\ 1 & \text{if } \text{Exp}(-\lambda_i) \le U_{1j} \le 1 \end{cases} \qquad (3.4)$$

and

$$T_{1j} = \begin{cases} 1 & \text{if } F_{1j} = 0 \\ \dfrac{-\ln(U_{1j})}{\lambda_1} & \text{if } F_{1j} = 1 \end{cases} \qquad j=1,2,\ldots,N_1 \qquad (3.5)$$

c. For the LG method, compute

$$A_1 = \frac{(2N_1 - 3)}{(2N_1 - 2)} \tag{3.6}$$

$$B_1 = \frac{N_1}{N_1 - 1} \tag{3.7}$$

and

$$F_1 = \sum_{j=1}^{N_1} F_{1j} \tag{3.8}$$

d. For the CLG method, compute the test time for component one.

$$Tsum_1 = \sum_{j=i}^{N_1} T_{1j} \tag{3.9}$$

Step three: Compute

$$\hat{T}_1 = A_1\left(\frac{F_1}{N_1}\right) + \frac{B_1}{2}\left(\frac{F_1}{N_1}\right)^2 \tag{3.10}$$

for the LG method, and

$$\hat{Z}_1 = \frac{F_1}{Tsum_1} \tag{3.11}$$

for the CLG method.

Step four: The component test times for the MLR method are defined differently from that of the CLG method. The MLR method uses only accumulated time to the last failure for each component; if no failures occur this is zero. Thus, compute for component one

$$T_1^* = \sum_{j=1}^{F_1} T_{1j}^* \qquad \text{if } F_1 > 0 \tag{3.12}$$

where T_{ij}^* is as defined on page 20.

Step five: Compute the number of trials on component two,

$$N_2 = N_1 - \left[\tfrac{1}{2}F_1\right]$$

Repeat steps two, three, and four for component two. Then compute $N_3 = N_1 - \left[\tfrac{1}{2}(F_1 + F_2)\right]$. In general, the sample size for the ith component is

$$N_i = N_1 - \left[\tfrac{1}{2} \sum_{j=1}^{i-1} F_j\right] \tag{3.13}$$

Step six: Continue alternating through steps two through five for the remainder of the components.

Step seven: Obtain

$$\hat{S} = \sum_{i=1}^{K} \hat{T}_i \tag{3.14}$$

$$\hat{Z} = \sum_{i=1}^{K} \hat{Z}_i \tag{3.15}$$

$$\widehat{Var}(\hat{S}) = \sum_{i=1}^{K} (\hat{T}_i/N_i) \tag{3.16}$$

$$\widehat{Var}(\hat{Z}) = \sum_{i=1}^{K} (\hat{Z}_i/N_i) \tag{3.17}$$

$$\hat{r} = \hat{S}^2 / \widehat{Var}(\hat{S}) \tag{3.18}$$

$$\hat{L} = \hat{Z}^2 / \widehat{Var}(\hat{Z}) \tag{3.19}$$

and the total number of replica failures

$$FT \equiv \sum_{i=1}^{K} F_i \tag{3.20}$$

Step eight: To preclude non-integer degrees of freedom, denote

$$[2\hat{r}] = \max\left\{2, \text{smallest integer} \geq 2\hat{r}\right\} \tag{3.21}$$

$$[2\hat{L}] = \max\left\{2, \text{smallest integer} \geq 2\hat{L}\right\} \tag{3.22}$$

Step nine: For the MLR procedure, compute

$$U = \begin{cases} \min N_i & \text{if } FT = 0 \\ \min T_i^* & \text{otherwise, } i=1,2,\ldots,K \end{cases} \tag{3.23}$$

and $$I_i = \begin{cases} \text{greatest index such that} \\ \quad \sum_{j=1}^{I_i} T_{ij}^* \leq U & \text{if } F_i > 0 \\ \quad 0 & \text{if } F_i = 0 \end{cases} \tag{3.24}$$

then obtain $$I = \max\left\{1, \sum_{i=1}^{K} I_i\right\} \tag{3.25}$$

The zero failure portion of the U computation sets U to the observed number of "system" successes (see Appendix D).

Step ten: Compute the replica one LCL's for the three procedures. Define RD, RT, and RL for discrimination as below

a. LG method

$$\hat{R}_{S,L(\alpha)} \equiv RD = \text{Exp}\left\{\frac{-[2\hat{r}]\hat{S}}{\chi^2_{1-\alpha,[2\hat{r}]}}\right\} \tag{3.26}$$

b. CLG method

$$\hat{R}_{S,L(\alpha)} \equiv RT = \text{Exp}\left\{\frac{-[2\hat{L}]\hat{z}}{\chi^2_{1-\alpha,[2\hat{L}]}}\right\} \tag{3.27}$$

c. MLR method

$$\hat{R}_{S,L(\alpha)} \equiv RL = \text{Exp}\left\{\frac{-\chi^2_{\alpha,2I}}{2U}\right\} \tag{3.28}$$

Step eleven: Replicate steps two through ten 500 times to obtain

$$RD_1, RD_2, \ldots\ldots\ldots\ldots\ldots, RD_{500}$$
$$RT_1, RT_2, \ldots\ldots\ldots\ldots\ldots, RT_{500}$$
$$RL_1, RL_2, \ldots\ldots\ldots\ldots\ldots, RL_{500}$$
$$FT_1, FT_2, \ldots\ldots\ldots\ldots\ldots, FT_{500}$$

Step twelve: Order the LCL's in step eleven.

$$RD_{(1)}, RD_{(2)}, \ldots\ldots\ldots\ldots, RD_{(500)}$$
$$RT_{(1)}, RT_{(2)}, \ldots\ldots\ldots\ldots, RT_{(500)}$$
$$RL_{(1)}, RL_{(2)}, \ldots\ldots\ldots\ldots, RL_{(500)}$$

Step thirteen: Next, compute $A_{1-\alpha}$ for the three procedures.

$$RD_{(500(1-\alpha))}$$

$$RT_{(500(1-\alpha))}$$

$$RL_{(500(1-\alpha))}$$

and the average number of failures per replica

$$\widehat{TT} = \frac{1}{500} \sum_{j=1}^{500} FT_j \qquad (3.29)$$

Each procedure's sample variance is computed by

$$s^2 = \frac{1}{499} \sum_{j=1}^{500} \left[(\hat{R}_{S,L(\alpha)})_j - (\overline{\hat{R}_{S,L(\alpha)}}) \right]^2 \qquad (3.30)$$

where $\overline{\hat{R}_{S,L(\alpha)}}$ denotes the average LCL over the 500 replicas. Now each procedure's accuracy can be determined by comparing $A_{1-\alpha}$ to true system reliability R_s, and relative dispersion by comparing s.

Reference 1, the original LG method paper, used the quantity

$$TT = \sum_{i=1}^{K} N_i Q_i \qquad (3.31)$$

as a measure of accuracy and the amount of testing necessary to assure sufficient accuracy of the LG procedure. As long as no component sample size is below ten, Table I shows recommended TT for the LG method (Reproduced from Ref. 1, p. 19).

TABLE I

Recommended TT for the Log-Gamma Method

100(1-α)%	Minimum TT	Reasonable TT
80	3	4
90	8	10
95	15	20

The simulation computes equation (3.29) for comparison with the above Table.

IV. THE INITIAL SIMULATION

A. CASE DESCRIPTION

The initial cases analyzed had true system reliabilities ranging from near .70 upwards and were composed of fifteen or forty components. The average number of failures per replica, $\hat{TT}$, ranged up to near 15. The first simulation used the formulas for $\hat{R}_{S,L(\alpha)}$ on page 28. The results are listed in Table III. Two cases were also ran with a Weibull failure distribution, vice exponential, to test the CLG and MLR procedures' sensitivity to the exponential assumption. Appendix E develops the PIT method for generating exponential and Weibull failure data. Table IV lists Weibull results.

B. RESULTS AND ANALYSIS

The LG method conformed to Reference 1 assertions in that performance improved with increasing $\hat{TT}$, although Ref. 1 did not generate component sample sizes in the same manner as this study. This should, however, have no effect on the results.

Small $\hat{TT}$, such as in cases one and three, resulted in conservative LG and CLG LCL's (ie., $A_{1-\alpha} < R_s$).

TABLE II

NOMENCLATURE FOR SIMULATION RESULTS TABLES

N_1	:	Sample size of 1st component; analogous to the number of systems tested
K	:	Number of series connected components
R_i	:	True reliability of the ith component
R_s	:	True independent series system reliability $R_s = \prod_{i=1}^{K} R_i$
min	:	Minimum number of failures in one replica
max	:	Maximum number of failures in one replica
$\hat{TT}$	:	Average number of failures per replica; approximation for $TT = \sum_{i=1}^{K} N_i Q_i$
s	:	Sample standard deviation of $\hat{R}_{S,L(\alpha)}$
$A_{1-\alpha}$	:	The $500(1-\alpha)$th order statistic of $\hat{R}_{S,L(\alpha)}$ for each of the three methods
LG	:	The Log-Gamma method
CLG	:	The Continuous Log-Gamma method
MLR	:	The Modified Lieberman-Ross method
RLG	:	The Randomized Log-Gamma method (modification three only)

For example, the case one sample size for the first component was $N_1 = 20$, and the series system modeled had $K = 15$ components each with true reliability $R_i = .99$. True system reliability $R_s = .860$. There were as few as zero (min = 0) and as many as eight (max = 8) component failures

TABLE III

Initial Simulation Results

CASE	$\hat{TT}$	R_s	80% LCL's $A_{1-\alpha}/s$			90% LCL's $A_{1-\alpha}/s$		
			LG	CLG	MLR	LG	CLG	MLR
Case 1: N_1=20 K=15, R_i=.99 i=1,15 min=0, max=8	2.9	.860	.799/.074	.802/.086	.851/.255	.680/.078	.769/.089	.853/.263
Case 2: N_1=50 K=15, R_i=.99 i=1,15 min=1, max=17	7.2	.860	.851/.052	.850/.054	.861/.253	.829/.050	.837/.054	.855/.264
Case 3: N_1=20 K=15, R_i=.99 i=1,14; R_{15}=.995 min=0, max=8	2.8	.864	.799/.073	.803/.085	.854/.254	.680/.078	.770/.088	.856/.263
Case 4: N_1=50 K=15, R_i=.99 i=1,14; R_{15}=.995 min=1, max=16	7.0	.864	.851/.051	.850/.053	.866/.251	.838/.049	.837/.054	.857/.262
Case 5: N_1=20 K=15, R_i=.995 i=1,14; R_{15}=.85 min=0, max=11	4.3	.792	.749/.096	.749/.104	.776/.224	.680/.093	.722/.104	.773/.230
Case 6: N_1=50 K=15, R_i=.995 i=1,14; R_{15}=.85 min=3, max=20	10.5	.792	.787/.058	.784/.059	.794/.225	.775/.058	.773/.059	.796/.235

TABLE III
(continued)

CASE	$\hat{TT}$	R_s	80% LCL's $A_{1-\alpha}/s$			90% LCL's $A_{1-\alpha}/s$		
			LG	CLG	MLR	LG	CLG	MLR
Case 7: N_1=20 K=15, R_1=.85 R_i=.99; i=2,15 min=0, max=13	5.5	.738	.703/.089	.698/.095	.718/.245	.677/.081	.676/.090	.708/.240
Case 8: N_1=50 K=15, R_i=.85 R_i=.99; i=2,15 min=5, max=29	13.7	.738	.730/.061	.728/.062	.732/.235	.733/.061	.730/.062	.746/.237
Case 10: N_1=50 K=40, R_i=.995 i=1,40 min=1, max=18	9.4	.818	.810/.056	.808/.058	.810/.227	.795/.056	.794/.058	.806/.278

TABLE IV

Initial Results: Weibull Sensitivity Analysis

CASE	$\hat{TT}$	R_s	80% LCL's $A_{1-\alpha}/s$			90% LCL's $A_{1-\alpha}/s$		
			LG	CLG	MLR	LG	CLG	MLR
Case 1 with $\beta = .5$ min=0, max=8	2.9	.860	.799/.074	.801/.087	.851/.266	.680/.078	.767/.090	.851/.271
Case 1 with $\beta = 3.0$ min=0, max=8	2.9	.860	.799/.074	.803/.085	.851/.233	.680/.078	.772/.088	.856/.249

over the 500 replicas of case one, with the average number of failures per replica being 2.9 ($\hat{TT} = 2.9$). Comparing $A_{1-\alpha}$ to R_s for accuracy, the LG 80% procedure had $A_{1-\alpha}$=.799 and s=.074, whereas the MLR had $A_{1-\alpha}$=.851 and s =.255. Thus, in case one both procedures were conservative, but the MLR was more accurate. However, the LG procedure had less than one-third the dispersion of the MLR procedure.

Increased testing, as in cases two and four, not only improved their accuracy but showed smaller sample standard deviation. The same comparison holds for case five with case six, and case seven with case eight. Also note the closeness of both $A_{1-\alpha}$ and s for the LG and CLG methods. This basically says that even though more assumptions are required and information is needed for the CLG method than the LG method, it yields no better results.

Case three of Ref. 1 is essentially case five of Table III; case five is essentially case six, and case eleven is essentially case two (note that Ref. 1 miscomputed TT in cases three and five). The accuracy achieved by case five was not as high as that of case three, but the sample standard deviation was comparable. In the other two comparable cases, both the accuracy and dispersion match closely.

The MLR procedure is the most accurate of the three, both in low and high $\hat{TT}$ cases. The sample standard deviation, however, is three to five times that of the LG method. The most disturbing thing about the MLR method is that even with increased testing ($\hat{TT}$), s is still high.

Reference 6 shows that the LR method is expected to have high dispersion above about ten observed failures (Figure 1, p. 22). The MLR method, even in the cases with ten or more failures, still has much higher dispersion than the LG and CLG procedures. Reference 6 attempted to improve the LR method by reduction of data loss. By ordering the failure times in increasing order, maximum data usage and chi square degrees of freedom were attained. The resultant LCL's were found to be biased in an unpredictable direction (based on a single parameter) and were even more widely dispersed. It was therefore proposed that all possible orderings of the data be considered, and research is currently being conducted in that area. Thus this study pursued it no further.

The two Weibull cases showed that the CLG and MLR methods were slightly sensitive to the exponential assumption. Since $T \sim$ exponential (λ) implies $T^{1/\beta} \sim$ Weibull (β, λ) (see Appendix E), then for $\beta = .5$ and $T < 1$ mission unit, the generated Weibull failure time is sooner than its exponential counterpart. Likewise, for $\beta = 3.0$, the Weibull time is later. In fact, case one with $\beta = .5$ gave slightly lower $A_{1-\alpha}$ for both methods (excepting the 80% MLR LCL). With $\beta = 3.0$, both cases gave slightly higher $A_{1-\alpha}$ (again, with the same exception as above).

V. MODIFICATIONS OF THE METHODS

A. MODIFICATION ONE: EXACT DEGREES OF FREEDOM AND ADJUSTMENT FOR ZERO FAILURES

1. Description

The original LG method employed only integer degrees of freedom; the initial results of Chapter IV adhered to this convention for all three methods. The accessibility of computers today makes chi square values with non-integer degrees of freedom readily available. Rounding up degrees of freedom by usage of the "smallest integer greater than or equal to" function can also cause monotonicity problems in $\hat{R}_{S,L(\alpha)}$. Consider LG 80% LCL's in the case where $N_1=20$, and only one failure occurs. Then

$$\hat{S} = \sum_{i=1}^{K=15} \hat{T}_i \qquad (5.1)$$

$$= .05$$

$$\hat{r} = \frac{\hat{S}^2}{\sum_{i=1}^{15} \frac{\hat{T}_i}{N_i}} \qquad (5.2)$$

$$= \frac{(.05)^2}{(.05/20)} = 1$$

then $2\hat{r} = 2$

$[2\hat{r}] = 2$

thus, $$\hat{R}_{S,L(\alpha)} = \text{Exp}\left\{ \frac{-[2\hat{r}]\hat{S}}{\chi^2_{1-\alpha,[2\hat{r}]}} \right\} \qquad (5.3)$$

$$\hat{R}_{S,L(.2)} = \text{Exp}\left\{\frac{-2(.05)}{\chi^2_{.8,2}}\right\} = .799$$

Now, take the exact same case and outcome, except that instead of one failure in one component, two failures occur in the same component. Then,

$$\hat{S} = .103$$

$$\hat{r} = \frac{(.103)^2}{(.103/20)} = 2.053$$

$$2\hat{r} = 4.105$$

$$\lceil 2\hat{r} \rceil = 5$$

It follows that,

$$\hat{R}_{S,L(.2)} = \text{Exp}\left\{\frac{-5(.103)}{\chi^2_{.8,5}}\right\} = .803$$

The above LCL is higher even though one more failure occurred, due to rounding up degrees of freedom $2\hat{r}$.

Another improvement felt necessary was an adjustment for the event of no failures for the LG method. The initial procedure here gives

$$\hat{S} = 0$$

and $\hat{R}_{S,L(\alpha)} = 1.00$

Therefore, the following was incorporated: Suppose there are no failures in one replica with N_1 initial component trials. Then set the number of failures on the first component as $F_1 = \frac{1}{2}$ and compute $2\hat{r}$ as if one failure had occurred (this effectively constrains $2\hat{r} \geq 2$). This ad hoc and simplistic adjustment will produce zero-failure LCL's as a function of N_1, and satisfies monotonicity requirements.

The simulation was again run with the adjustments above for certain test cases. The results are listed in Table VI.

2. Observations and Analysis

Due to the fact that the MLR method never assumes non-integer degrees of freedom, it was unaffected by this modification. Both cases tested had slightly lower $A_{1-\alpha}$ and sample standard deviation, s for the LG and CLG methods. The smaller s can be attributed to less discreteness in $\hat{R}_{S,L(\alpha)}$. The lower $A_{1-\alpha}$ is caused by the fact that $\chi^2_{1-\alpha,2n}$ increases faster than 2n. By rounding up the degrees of freedom 2n to [2n], the initial simulation produced slightly higher LCL's than did modification one.

TABLE V

Behavior of $\frac{2n}{\chi^2_{1-\alpha,2n}}$ for $\alpha = .2$

2n	$\chi^2_{.8,2n}$	$\frac{2n}{\chi^2_{.8,2n}}$
2	.446	4.48
3	1.01	2.97
5	2.34	2.14
10	6.18	1.62
30	23.4	1.28

TABLE VI

SIMULATION RESULTS WITH MODIFICATION ONE

a) Exact Degrees of Freedom

b) A Zero-Failure Adjustment for the Log-Gamma Method

CASE	$\hat{TT}$	R_s	80% LCL's $A_{1-\alpha}/s$			90% LCL's $A_{1-\alpha}/s$		
			LG	CLG	MLR	LG	CLG	MLR
Case 1: N_1=20 K=15, R_i=.99 i=1,15 min=0, max=8	2.9	.860	.783/.064	.779/.077	.851/.255	.680/.050	.679/.080	.853/.263
Case 5: N_1=20 K=15, R_i=.995 i=1,14; R_{15}=.85 min=0, max=8	4.3	.792	.740/.088	.739/.098	.776/.224	.680/.074	.679/.093	.773/.230

While this modification slightly decreased dispersion, it did not improve the accuracy of the LG and CLG methods (and had no effect on the MLR method). The only real gain was to solve an infrequent monotonicity problem. If only integer degrees of freedom are available, it is recommended that a more refined scheme of rounding non-integer outcomes be devised. For instance, in the earlier example, $2\hat{r} = 4.105$ should be rounded down to 4 vice up to 5. The LG LCL becomes .779, which is lower than the one-failure .799 outcome, but still higher than the .738 three failure outcome (see Table VII).

Being that the CLG method performs no better than the LG method, but requires more assumptions and information, no further effort was expended on it. All subsequent modifications are directed towards improving the accuracy of the LG method while maintaining small dispersion.

B. MODIFICATION TWO

1. An Accuracy Improvement, and an Adjustment for the Degrees of Freedom in the 90% Lower Confidence Limit Procedure

The primary weakness of the LG method is its poor performance when few failures occur. More testing, and thereby more failures, results in a more accurate $A_{1-\alpha}$. Table VII lists and plots possible LG LCL's for zero through ten failures when $N_1 = 20$. Note the gaps caused by discreteness up to one failure. Also note that the 90% LCL for two and three failures is higher than that for one failure.

TABLE VII

LOG GAMMA 80 AND 90% LCLS10-10 FAILURES

N1= 20. ONLY ONE FAILURE PER COMPONENT

FAILURES	PHAT	80%LCL	90%LCL
0	2.0300	0.8914	0.7838
1	2.0300	0.7293	0.6221
2	3.9514	0.7295	0.6796
3	5.9966	0.7382	0.6551
4	7.9489	0.6926	0.6165
5	9.9645	0.6496	0.5781
6	11.9677	0.6064	0.5381
7	13.9963	0.5665	0.5011
8	15.9252	0.5266	0.4637
9	17.9025	0.4899	0.4296
10	19.8510	0.4534	0.3956

80% LCL("·") AND 90% LCL("•") VS. FAILURES

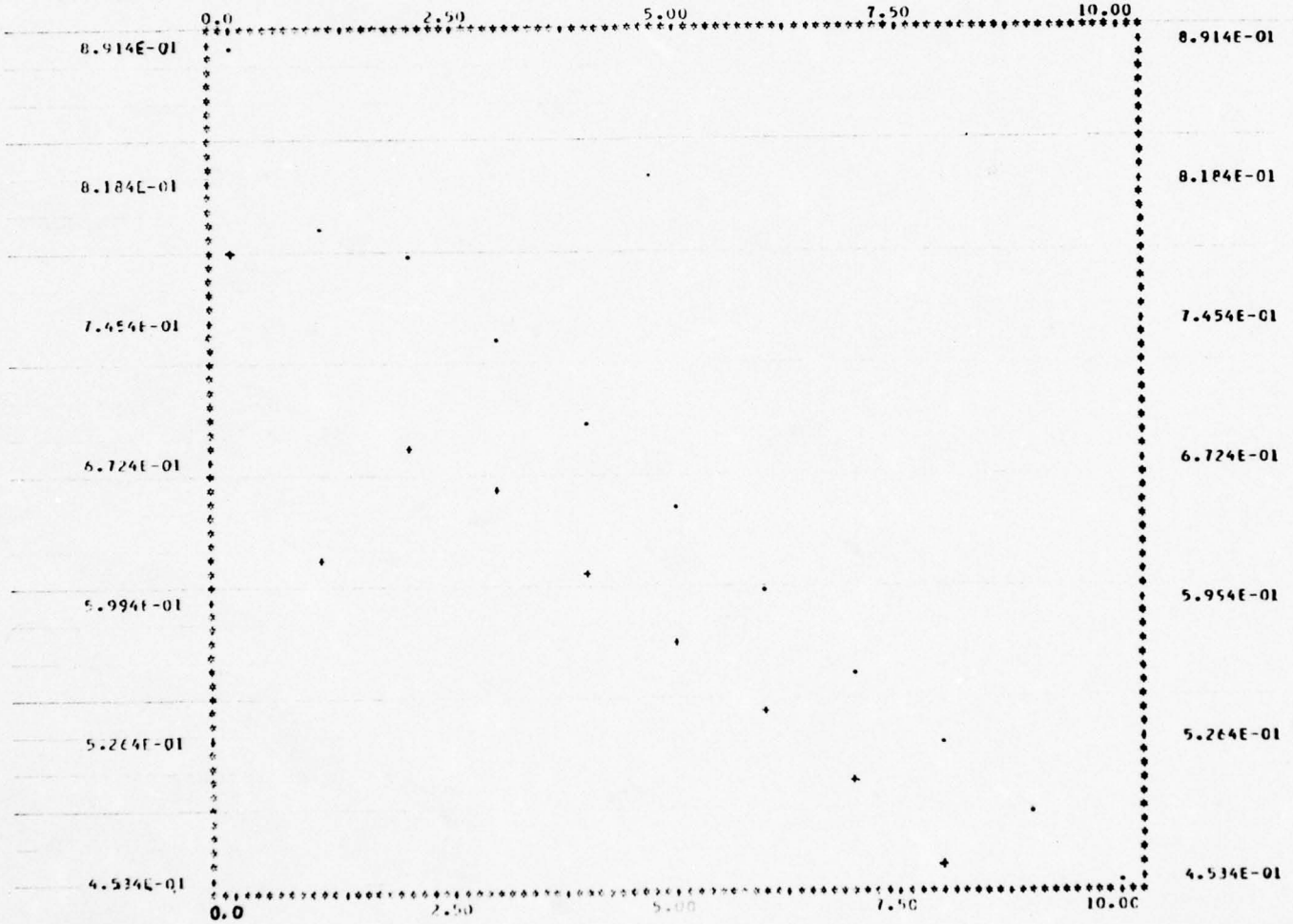


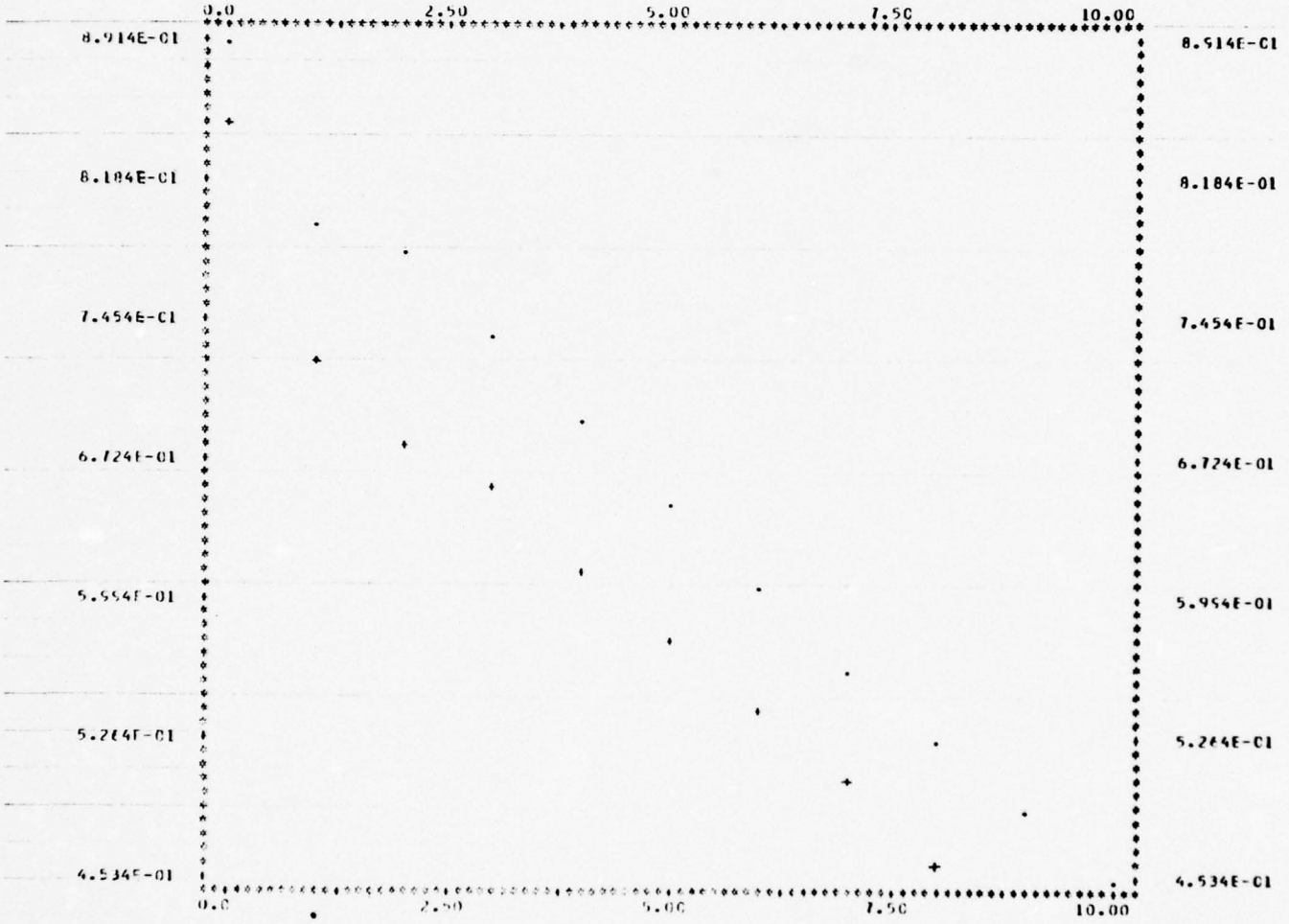
TABLE VIII

LOG GAMMA 80 AND 90Z LCLS10-10 FAILURES

NI= 20, ONLY ONE FAILURE PER COMPONENT

FAILURES	2RMAT	80%LCL	90%LCL	2RMAT-90%LCL
0	2.0000	0.8914	0.8450	2.5000
1	2.0000	0.7793	0.7202	2.5000
2	3.9774	0.7795	0.6796	3.9774
3	5.9966	0.7392	0.6551	5.9966
4	7.9889	0.6926	0.6165	7.9889
5	9.9845	0.6496	0.5781	9.9845
6	11.9677	0.6064	0.5381	11.9677
7	13.9563	0.5665	0.5011	13.9563
8	15.9252	0.5266	0.4637	15.9252
9	17.9025	0.4899	0.4296	17.9025
10	19.8510	0.4534	0.3956	19.8510

80% LCL(".") AND 90% LCL("*") VS. FAILURES



This non-monotonicity is caused by the sharp drop-off in $\chi^2_{1-\alpha} = \chi^2_{.9}$ at low degrees of freedom. As an adjustment, $2\hat{r}$ for 90% LG LCL's was constrained to be at least 2.5; the minimum $2\hat{r}$ for 80% LCL's remains 2.0. The results can be seen in Table VIII.

Consider the classical binomial component LCL procedure on the probability of mission success p. If p is estimated by $\hat{p} = X/N$ where X is the observed number of successes in N trials, then a 100(1-α)% LCL on p, denoted $\hat{p}_{L(\alpha)}$, is the solution for p in the equation

$$\sum_{j=x}^{N} \binom{N}{j} p^j (1 - p)^{N-j} = \alpha \qquad (5.4)$$

Under the simulation of Chapter III, if N_1 is the component one sample size and no failures occur in a replica, it can be equivalently stated that N_1 "systems" were tested without failure. Likewise, one component failure implies exactly one system failure. However, when two or more different types of components fail, it cannot be ascertained how many system failures occurred. Figure 3 illustrates this for the two-failure case.

Figure 3

Component Failures Versus System Failures

Component	1	2	.....	K			1	2	.....	K
N_1 systems tested with one system failure	x	x	.....	-	or	N_1 systems tested with two system failures ?	x	-	.....	-
	-	-	.....	-			-	x	.....	-
	⋮						⋮			
	-	-	.....	-			-	-	.....	-

Treating a "system" as a component, equation (5.4) can be used to calculate $\hat{R}_{S,L(\alpha)}$ when zero or one failures occur.

a. Zero failures:

For X=N successes, $\hat{p}_{L,(\alpha)} = \alpha^{1/N_1}$. Modification two revised this upwards to

$$\hat{R}_{S,L(\alpha)} = \sqrt[N_1]{\alpha} + .2(1 - \sqrt[N_1]{\alpha}) \tag{5.5}$$

b. One failure:

Equation (5.4) was applied directly for $X=N_1-1$. That is, $\hat{R}_{S,L(\alpha)}$ is the solution for p in the equation

$$\sum_{j=N_1-1}^{N_1} \binom{N_1}{j} p^j (1-p)^{N_1-j} = \alpha \tag{5.6}$$

or $$N_1 p^{N_1-1}(1-p) + p^{N_1} = \alpha \tag{5.7}$$

c. Two through eight failures:

The LG method was adjusted upwards as follows:

$$\hat{R}'_{S,L(\alpha)} = \tfrac{1}{2}\left[\hat{R}_{S,L(\alpha)}(N_1,FT) + \hat{R}_{S,L(\alpha)}(N_1,FT-1)\right] \tag{5.8}$$

where $\hat{R}_{S,L(\alpha)}(N_1,FT)$ denotes the LG LCL computed with the observed total replica failures FT when the original sample size is N_1.

d. More than eight failures:

The original LG procedure is used with exact degrees of freedom. The results are in Table IX.

TABLE IX

Modification Two of the Log-Gamma Method

-use of the classical binomial model for zero and one failure
-use of the averaged LG method between two and eight failures
-an adjustment for degrees of freedom in the 90% procedure

CASE	$\hat{TT}$	R_s	80% LCL's $A_{1-\alpha}/s$			90% LCL's $A_{1-\alpha}/s$		
			LG	CLG	MLR	LG	CLG	MLR
Case 1: N_1=20 K=15, R_i=.99 i=1,15 min=0, max=8	2.9	.860	.792/.078	.779/.077	.851/.255	.855/.093	.679/.080	.853/.263
Case 5: N_1=20 K=15, R_i=.995 i=1,14; R_{15}=.85 min=0, max=11	4.3	.792	.759/.095	.739/.098	.776/.224	.700/.094	.679/.093	.773/.230
Case 10: N_1=20 K=15, R_i=.995 i=1,15 min=0, max=6	1.5	.928	.938/.075	1.00/.100	.923/.251	.913/.102	1.00/.148	.891/.267

2. Observations and Analysis

Modification two did improve the accuracy of the LG method, while slightly increasing dispersion. The 90% procedure was noticeably improved by the degrees of freedom constraint, although still conservative. There is, however, still too much discreteness in values of $\hat{R}_{S,L(\alpha)}$ when few failures occur. Table X generated at NPS by APL subroutine "Histlist" (see documentation), illustrates this for case five. This discreteness usually causes the procedure to yield more than the target $1-\alpha$ confidence, and is a problem with nearly any method using only attributes data.

This being the case, the next logical step was to fit a continuous curve to the procedure by randomization.

C. MODIFICATION THREE: RANDOMIZATION

1. Randomization of the Log-Gamma Method

Many LCL procedures which use only attributes data are degraded in that they yield conservative bounds. Because only a finite set of outcomes can be observed for any test plan, then only a finite set of LCL's can be realized. Thus, $\hat{R}_{S,L(\alpha)}$ is a discrete random variable which ideally gives the LCL closest to but not greater than the "exact" LCL.

The 1975 article of Barr and Jayachandran [Ref. 7, p. 67-68] developed an exact, randomized binomial component LCL procedure. The same rationale used before (see Figure 3) allows the randomized binomial procedure to be used for

TABLE X

Case 5 Log-Gamma 80 Percent LCL's with Modification Two

SERIAL NUMBER	ORDERED DATA	OCCURENCES	PERCENT
1	.407610	1	.002
2	.444490	1	.002
3	.444960	1	.002
4	.471830	1	.002
5	.475840	2	.004
7	.477390	3	.006
10	.482700	2	.004
12	.484660	1	.002
13	.487430	1	.002
14	.527610	1	.002
15	.534450	3	.006
18	.538120	10 *	.020
28	.538550	1	.002
29	.540650	1	.002
30	.541090	1	.002
31	.542420	2	.004
33	.542860	1	.002
34	.571020	11 *	.022
45	.576210	1	.002
46	.577770	7 *	.014
53	.578300	2	.004
55	.580530	10 *	.020
65	.581070	1	.002
66	.583950	1	.002
67	.584260	2	.004
69	.584910	2	.004
71	.616160	13 **	.026
84	.620600	5 *	.010
89	.622570	14 **	.028
103	.623270	1	.002
104	.624390	24 ***	.048
128	.625090	2	.004
130	.627950	1	.002
131	.628010	1	.002
132	.628080	1	.002
133	.662770	23 ***	.045
156	.666380	1	.002
157	.666490	14 **	.028
171	.667840	1	.002
172	.668570	16 **	.032

TABLE X
(continued)

SERIAL NUMBER	ORDERED DATA	OCCURENCES		PERCENT
188	.669380	21	***	.042
209	.669530	1		.002
210	.670340	6	*	.012
216	.671090	2		.004
218	.672070	1		.002
219	.673220	1		.002
220	.710230	37	****	.074
257	.713320	22	***	.044
279	.714870	25	***	.050
304	.715380	6	*	.012
310	.716280	2		.004
312	.756600	47	******	.094
359	.758850	21	***	.042
380	.759200	21	***	.042
401	.761320	2		.004
403	.789380	39	*****	.078
442	.791840	27	***	.054
469	.890140	21	***	.042
490	.938140	11	*	.022

"system" LCL's when zero or one failure occurs. Equations for the Randomized Log-Gamma method (RLG) are derived in Appendix F. An example is given in Appendix G.

a. Randomized Log-Gamma Procedure

(1) If zero failures occur, generate a uniform (0,1) random number Y. Then

$$\hat{R}_{S,L(\alpha)} = \begin{cases} \left(\frac{\alpha}{1-Y}\right)^{1/N_1} & \text{if } Y \leq 1-\alpha \\ 1.0 & \text{if } Y > 1-\alpha \end{cases} \tag{5.9}$$

(2) When one failure is observed, generate a uniform (0,1) random number Y. Then $\hat{R}_{S,L(\alpha)}$ is the solution for p in the equation

$$(1-Y)N_1 p^{N_1-1}(1-p) + p^{N_1} = \alpha \quad \text{if } Y \leq 1-\alpha \tag{5.10}$$

This solution can be approximated by

$$\hat{R}_{S,L(\alpha)} = \alpha^{1/N_1} \tag{5.11}$$

for $Y > 1-\alpha$ (see Appendix F).

(3) Two through four failures inclusive:

Generate a uniform (0,1) random number Y_i for each component that had at least one failure, i=1,2,...,K. Then use the LG method with $F_i^* = F_i - Y_i$ instead of F_i. Thus, the RLG formulas become

$$A_i = \frac{2N_i - 3}{2N_i - 2} \tag{5.12}$$

$$B_i = \frac{N_i}{N_i - 1} \tag{5.13}$$

$$\hat{T}_i = A_i\left(\frac{F_i - Y_i}{N_i}\right) + \frac{B_i}{2}\left(\frac{F_i - Y_i}{N_i}\right)^2 \qquad (5.14)$$

$$\hat{S} = \sum_{i=1}^{K} \hat{T}_i \qquad (5.15)$$

$$\hat{r} = \frac{\hat{S}^2}{\sum_{i=1}^{K} \frac{\hat{T}_i}{N_i}} \qquad (5.16)$$

and $$\hat{R}_{S,L(\alpha)} = \mathrm{Exp}\left\{\frac{-2\hat{r}\hat{S}}{\chi^2_{1-\alpha,2\hat{r}}}\right\} \qquad (5.17)$$

The only differences between the original LG formulas and those above is the computation of $\hat{T}_i$ in equation (5.14), and usage of exact degrees of freedom.

(4) Five or more failures:

Use the non-randomized LG method with exact degrees of freedom. That is, use the formulas in (3) above except use F_i vice $F_i - Y_i$ in the computation of $\hat{T}_i$.

2. Modification Three Cases

The number of cases simulated under modification three was expanded greatly. Cases one through twelve have an N_1 of either 20 or 50, and cover the range of pertinent system reliability R_s. All twelve cases were first ran with the number of components K=15, and then reran with K=40. Cases with equal component reliabilities were also ran with K=4. The system sample size was reduced to ten, and the applicable twelve cases ran again, with K = 15, 40, and 4. Modification three results are listed in Table XI

for K = 15, Table XII for K = 40, and Table XIII for K = 4. Table XIV lists results of cases from Ref. 1, which had highly-varied component sample sizes.

3. Results, Analysis, and Conclusions

All three method results were tabled for comparison under modification three, although randomization of the LG method was the only change.

The accuracy of the RLG was comparable to that of the MLR, while having much lower sample standard deviations. Both the RLG and CLG methods had dispersion inversely proportional to sample size, while the MLR continued to consistantly have high dispersion regardless of the circumstances.

The most significant improvement in the RLG was its performance with few failures and less testing. Especially in the K = 4 and K = 15 cases with small $\hat{T}T$, the RLG was near exact while the CLG sometimes gave $A_{1-\alpha}$ of 1.00 (meaning at least the upper (100)α% of the simulated distribution was LCL's of 1.00). The MLR procedure was inaccurate for very high reliability systems, such as case 11 with K = 15, due to the upper limit of its possible LCL values. In case 11, with $N_1 = 20$, the maximum MLR LCL is .923 with zero failures (see Appendix D), whereas true system reliability is .956. As in the earlier LG results, the 90% RLG procedure was usually less accurate, less stable and slightly more dispersed. The presence of one component with dissimilar

reliability, like case seven with K = 15 seemed to degrade accuracy somewhat. Additional testing, however, as in case eight, dampens this out.

Cases one and two of the original LG paper [Ref. 1] were also simulated with modification three to check sensitivity to mixed sample sizes (see Table XIV). Note that randomization was rarely necessary in case two (min = 2), and never in case one (min = 5). The accuracy of all three methods was high due to the large amount of testing. Note the small sample size of the twelth component in case one. This seemingly had little effect on the RLG method's accuracy, although s is suspiciously large when compared to case two. When R_{12} was set to .85, the effect on the RLG became clearer. When a component twelve failure occurred, its contribution to $\hat{S}$ and $\hat{r}$ so dominates that $\hat{R}_{S,L(\alpha)}$ is nearly halved from what it would have been otherwise.

Figures 4 and 5 give histograms for $\hat{R}_{S,L(\alpha)}$ of the RLG procedure generated under cases one and two respectively. Both have skewness and kurtosis roughly comparable to a normal distribution, and both have medians near-equal to their medians (symmetric). Figure 5 (case two with $N_1 = 50$) has a lower maximum $\hat{R}_{S,L(\alpha)}$ and a higher minimum than Figure 4 (case one with $N_1 = 20$), reflecting the decreased dispersion of the RLG procedure with increased testing.

TABLE XI

Modification Three Results: The Randomized Log-Gamma Method

K = 15

CASE	$\hat{TT}$	R_s	80% LCL's $A_{1-\alpha}/s$			90% LCL's $A_{1-\alpha}/s$		
			RLG	CLG	MLR	RLG	CLG	MLR
Case 1: N_1=20 K=15, R_i=.99 i=1,15 min=0, max=8	2.9	.860	.872/.088	.799/.080	.853/.283	.852/.103	.679/.084	.864/.286
Case 2: N_1=50 K=15, R_i=.99 i=1,15 min=0, max=15	7.2	.860	.847/.060	.847/.052	.865/.258	.854/.056	.829/.051	.864/.268
Case 1T: N_1=10 K=15, R_i=.99 i=1,15 min=0, max=6	1.4	.860	.861/.123	1.00/.172	.851/.289	.855/.178	1.00/.238	.794/.287
Case 3: N_1=20 K=15, R_i=.99 i=1,14; R_{15}=.995 min=0, max=8	2.8	.864	.869/.089	.800/.082	.866/.270	.860/.105	.679/.090	.870/.287
Case 4: N_1=50 K=15, R_i=.99 i=1,14; R_{15}=.995 min=0, max=15	6.9	.864	.848/.061	.847/.052	.862/.246	.855/.057	.829/.051	.874/.259
Case 3T: N_1=10 K=15, R_i=.99 i=1,14; R_{15}=.995 min=0, max=5	1.4	.864	.864/.122	1.00/.169	.851/.280	.856/.176	1.00/.238	.794/.279

TABLE XI
(continued)

CASE	$\hat{TT}$	R_s	80% LCL's $A_{1-\alpha}/s$			90% LCL's $A_{1-\alpha}/s$		
			RLG	CLG	MLR	RLG	CLG	MLR
Case 5: N_1=20 K=15, R_i=.995 i=1,14; R_{15}=.85 min=0, max=12	4.3	.792	.783/.100	.736/.083	.776/.225	.685/.090	.659/.072	.765/.229
Case 6: N_1=50 K=15, R_i=.995 i=1,14; R_{15}=.85 min=3, max=22	10.6	.792	.785/.055	.784/.056	.795/.212	.770/.055	.768/.056	.793/.220
Case 5T: N_1=10 K=15, R_i=.995 i=1,14; R_{15}=.85 min=0, max=7	2.1	.792	.794/.137	.642/.153	.790/.251	.794/.169	.471/.182	.793/.248
Case 7: N_1=20 K=15, R_i=.99 i=2,15; R_1=.85 min=1, max=11	5.6	.738	.762/.103	.690/.087	.717/.247	.676/.088	.655/.077	.719/.244
Case 8: N_1=50 K=15, R_1=.85 R_i=.99, i=2,15 min=3, max=26	13.7	.738	.730/.064	.728/.065	.757/.223	.731/.063	.728/.064	.751/.224
Case 7T: N_1=10 K=15, R_1=.85 R_i=.99; i=2,15 min=0, max=8	2.8	.738	.747/.143	.639/.132	.716/.250	.749/.155	.466/.131	.723/.241

TABLE XI
(continued)

CASE	$\hat{TT}$	R_s	80% LCL's $A_{1-\alpha}/s$			90% LCL's $A_{1-\alpha}/s$		
			RLG	CLG	MLR	RLG	CLG	MLR
Case 9: N_1=20 K=15, R_i=.995 i=1,15 min=0, max=5	1.5	.928	.933/.076	1.00/.100	.923/.257	.934/.117	1.00/.148	.891/.270
Case 10: N_1=50 K=15, R_i=.995 i=1,15 min=0, max=11	3.7	.928	.925/.050	.906/.042	.927/.237	.934/.053	.859/.041	.926/.252
Case 9T: N_1=10 K=15, R_i=.995 i=1,15 min=0, max=4	.8	.928	.932/.113	1.00/.192	.851/.205	.926/.159	1.00/.293	.794/.216
Case 11: N_1=20 K=15, R_i=.997 i=1,15 min=0, max=5	.9	.956	.955/.064	1.00/.107	.923/.245	.950/.103	1.00/.176	.891/.261
Case 12: N_1=50 K=15, R_i=.997 i=1,15 min=0, max=8	2.2	.956	.958/.039	.915/.040	.956/.221	.957/.053	.860/.052	.955/.245
Case 11T: N_1=10 K=15, R_i=.997 i=1,15 min=0, max=3	.4	.956	.952/.096	1.00/.177	.851/.196	.964/.132	1.00/.282	.794/.204

TABLE XII

Modification Three Results: The Randomized Log-Gamma Method

K = 40

CASE	$\hat{TT}$	R_s	80% LCL's $A_{1-\alpha}/s$			90% LCL's $A_{1-\alpha}/s$		
			RLG	CLG	MLR	RLG	CLG	MLR
Case 1: N_1=20 K=40, R_i=.99 i=1,40 min=1, max=15	7.2	.669	.650/.114	.645/.099	.664/.280	.662/.097	.613/.090	.668/.257
Case 2: N_1=50 K=40, R_i=.99 i=1,40 min=7, max=28	18.0	.669	.659/.062	.656/.063	.685/.266	.658/.061	.655/.062	.686/.252
Case 1T: N_1=10 K=40, R_i=.99 i=1,40 min=0, max=8	3.5	.669	.675/.159	.591/.123	.648/.276	.690/.148	.449/.105	.660/.251
Case 3: N_1=20 K=40, R_i=.99 i=1,39; R_{40}=.995 min=1, max=15	7.1	.672	.650/.114	.645/.099	.667/.281	.663/.098	.613/.090	.678/.259
Case 4: N_1=50 K=40, R_i=.99 i=1,39; R_{40}=.995 min=7, max=28	17.7	.672	.659/.063	.657/.063	.689/.265	.658/.062	.656/.062	.688/.252
Case 3T: N_1=10 K=40, R_i=.99 i=1,39; R_{40}=.995 min=0, max=9	3.5	.672	.665/.156	.592/.129	.660/.273	.683/.147	.450/.115	.671/.250

TABLE XII
(continued)

CASE	$\widehat{TT}$	R_s	80% LCL's $A_{1-\alpha}/s$			90% LCL's $A_{1-\alpha}/s$		
			RLG	CLG	MLR	RLG	CLG	MLR
Case 5: N_1=20 K=40, R_i=.995 i=1,39; R_{40}=.85 min=1, max=13	6.5	.699	.712/.111	.683/.094	.701/.245	.669/.097	.617/.086	.711/.240
Case 6: N_1=50 K=40, R_i=.995 i=1,39; R_{40}=.85 min=6, max=28	15.8	.699	.693/.063	.691/.064	.703/.249	.693/.063	.690/.063	.708/.243
Case 5T: N_1=10 K=40, R_i=.995 i=1,39; R_{40}=.85 min=0, max=8	3.2	.699	.733/.163	.595/.146	.680/.259	.729/.163	.451/.144	.675/.245
Case 7: N_1=20 K=40, R_1=.85 i=1,39; R_{40}=.85 min=0, max=8	9.4	.574	.567/.101	.559/.098	.569/.233	.539/.091	.534/.090	.556/.211
Case 8: N_1=50 K=40, R_1=.85 R_i=.99; i=2,40 min=14, max=35	23.9	.574	.573/.062	.568/.063	.551/.230	.571/.061	.565/.062	.567/.209
Case 7T: N_1=10 K=40, R_1=.85 R_i=.99; i=2,40 min=0, max=10	4.7	.574	.614/.162	.521/.126	.572/.247	.451/.130	.412/.103	.561/.221

TABLE XII
(continued)

CASE	$\hat{TT}$	R_s	80% LCL's $A_{1-\alpha}/s$			90% LCL's $A_{1-\alpha}/s$		
			RLG	CLG	MLR	RLG	CLG	MLR
Case 9: N_1=20 K=40, R_i=.995 i=1,40 min=0, max=10	3.9	.818	.811/.109	.777/.089	.830/.288	.830/.107	.679/.083	.822/.287
Case 10: N_1=50 K=40, R_i=.995 i=1,40 min=1, max=20	9.4	.818	.807/.057	.806/.056	.823/.273	.790/.056	.789/.055	.813/.275
Case 9T: N_1=10 K=40, R_i=.995 i=1,40 min=0, max=6	1.9	.818	.851/.135	.643/.161	.816/.287	.824/.187	1.00/.204	.794/.281
Case 11: N_1=20 K=40, R_i=.997 i=1,40 min=0, max=8	2.3	.887	.901/.089	.800/.093	.889/.285	.896/.117	.685/.115	.891/.293
Case 12: N_1=50 K=40, R_i=.997 i=1,40 min=1, max=13	5.8	.887	.909/.058	.867/.046	.878/.265	.858/.052	.847/.044	.880/.279
Case 11T: N_1=10 K=40, R_i=.997 i=1,40 min=0, max=6	1.1	.887	.896/.122	1.00/.188	.851/.254	.887/.176	1.00/.272	.794/.260

TABLE XIII

Modification Three Results: The Randomized Log-Gamma Method
K=4

CASE	$\hat{TT}$	R_s	80% LCL's $A_{1-\alpha}/s$			90% LCL's $A_{1-\alpha}/s$		
			RLG	CLG	MLR	RLG	CLG	MLR
Case 1: N_1=20 K=4, R_i=.99 i=1,4 min=0, max=4	.8	.961	.954/.062	1.00/.106	.923/.231	.950/.101	1.00/.177	.891/.246
Case 2: N_1=50 K=4, R_i=.99 i=1,4 min=0, max=7	2.0	.961	.961/.035	.915/.039	.960/.163	.959/.051	1.00/.054	.955/.185
Case 1T: N_1=10 K=4, R_i=.99 i=1,4 min=0, max=3	.4	.961	.958/.098	1.00/.178	.851/.200	.936/.123	1.00/.285	.794/.208
Case 9: N_1=20 K=4, R_i=.995 i=1,4 min=0, max=3	.4	.980	.979/.051	1.00/.097	.923/.133	.978/.073	1.00/.171	.891/.148
Case 10: N_1=50 K=4, R_i=.995 i=1,4 min=0, max=5	1.0	.980	.982/.030	1.00/.046	.968/.147	.982/.049	1.00/.079	.955/.168

TABLE XIII
(continued)

CASE	$\hat{TT}$	R_s	80% LCL's $A_{1-\alpha/s}$			90% LCL's $A_{1-\alpha/s}$		
			RLG	CLG	MLR	RLG	CLG	MLR
Case 9T: N_1=10 K=4, R_i=.995 i=1,4 min=0, max=2	.2	.980	.998/.082	1.00/.150	.851/.122	.996/.102	1.00/.245	.794/.134
Case 11: N_1=20 K=4, R_i=.997 i=1,4 min=0, max=3	.2	.988	.982/.042	1.00/.083	.923/.119	.984/.054	1.00/.150	.891/.132
Case 12: N_1=50 K=4, R_i=.997 i=1,4 min=0, max=12	.6	.988	.988/.024	1.00/.044	.968/.135	.985/.037	1.00/.082	.955/.150
Case 11T: N_1=10 K=4, R_i=.997 i=1,4 min=0, max=3	.1	.988	.977/.073	1.00/.121	.851/.116	.973/.089	1.00/.197	.794/.125

TABLE XIV

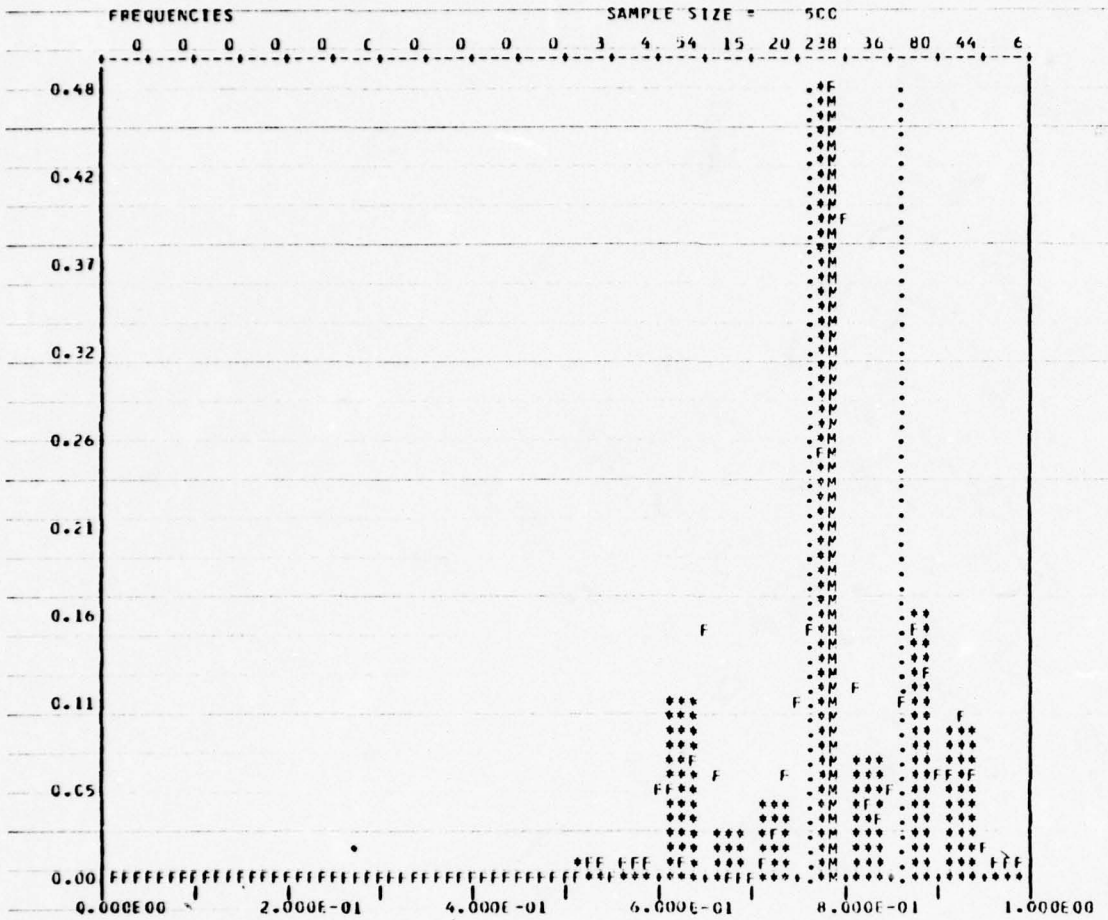
CASE	$\hat{TT}$	R_s	80% LCL's $A_{1-\alpha}/s$			90% LCL's $A_{1-\alpha}/s$		
			RLG	CLG	MLR	RLG	CLG	MLR
Case 1 of original Log-Gamma paper N_1=150 R_1=.995 N_2=90 R_2=.985 N_3=75 R_3=.979 N_4=100 R_4=.988 N_5=125 R_5=.982 N_6=18 R_6=.980 N_7=28 R_7=.967 N_8=125 R_8=.995 N_9=63 R_9=.970 N_{10}=125 R_{10}=.995 N_{11}=59 R_{11}=.968 N_{12}=5 R_{12}=.980 N_{13}=19 R_{13}=.900 K=13 min=5, max=26	15.3	.723	.722/.105	.720/.109	.709/.243	.723/.117	.720/.121	.722/.240

TABLE XIV
(continued)

CASE	$\hat{TT}$	R_s	80% LCL's $A_{1-\alpha}/s$			90% LCL's $A_{1-\alpha}/s$		
			RLG	CLG	MLR	RLG	CLG	MLR
Case 1 of original Log-Gamma paper except R_{12}=.85 min=6, max=31	16.5	.627	.673/.153	.670/.153	.617/.237	.676/.169	.673/.176	.639/.228
Case 2 of original Log-Gamma paper K=14, i=1,14 R_i=.99 N_1=250 N_3=120 N_5=130 N_7=70 N_9=100 N_{11}=60 N_{13}=20 N_2=40 N_4=15 N_6=65 N_8=75 N_{10}=90 N_{12}=60 N_{14}=30 min=2, max=20	10.2	.869	.870/.063	.869/.063	.868/.242	.866/.071	.865/.072	.866/.256

Figure 4

CASE 1 LOG-GAMMA 80 PERCENT LCLS WITH MODIFICATION THREE

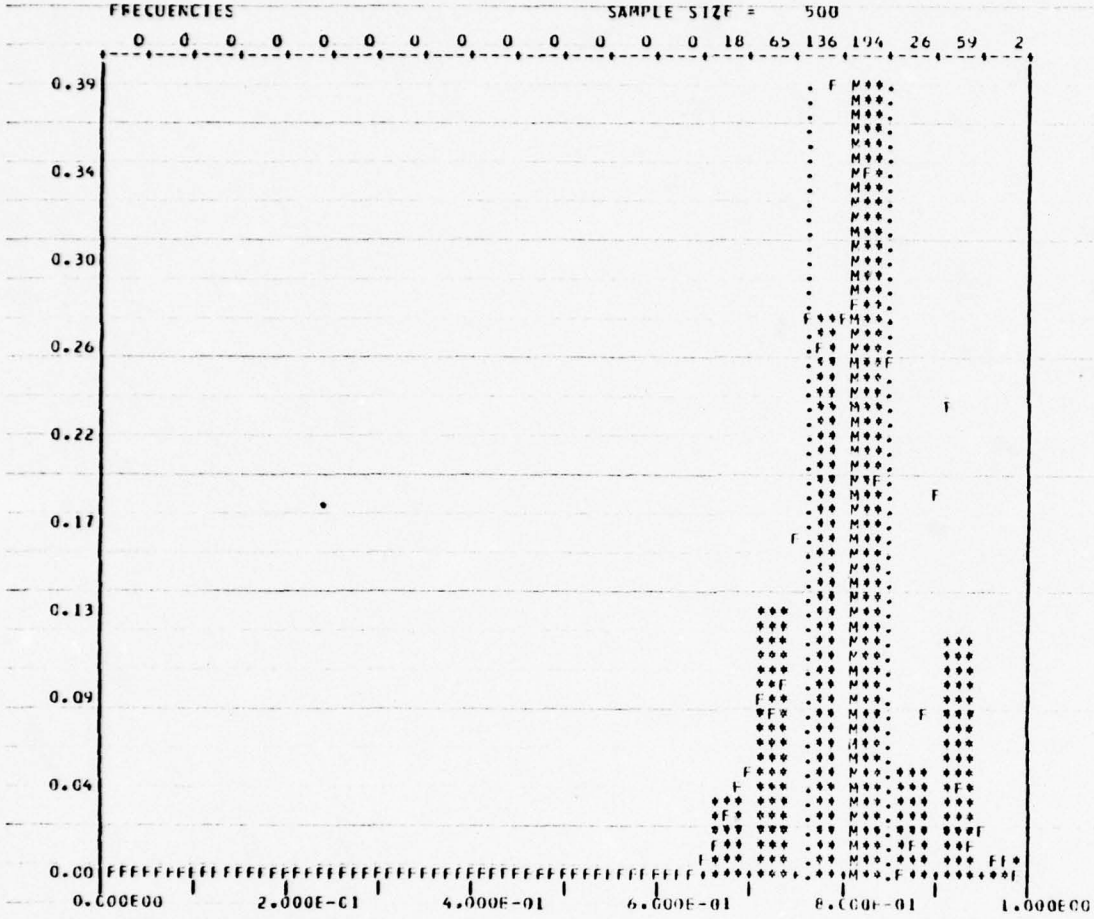


CELL WIDTH = 5.000000E-02

CENTRAL TENDENCY		SPREAD		HIGHER CENTRAL MOMENTS		DISTRIBUTION		
MEAN	7.903C55E-01	VARIANCE	7.773541E-03	M3	-3.378503E-04	MINIMUM		5.248000E-01
MEDIAN	7.921650E-01	STD DEV	8.816769E-02	M4	1.588500E-04	.10 QUANTILE		6.496200E-01
TRIMEAN	8.015863E-01	COEF VAR	1.115615E-01	SKEWNESS	-4.529421E-01	.25 QUANTILE	(HINGE)	7.632150E-01
MIDMEAN	7.9376C7E-01	MEAN DEV	6.157446E-02	KURTOSIS	2.907100E-01	.50 QUANTILE	(MEDIAN)	7.921650E-01
MIDRANGE	7.624000E-01	RANGE	4.752000E-01	BETA1	3.358259E-04	.75 QUANTILE	(HINGE)	8.588000E-01
CELM MEAN	7.8512C8E-01	MIDSPREAD	9.5585C0E-02	BETA2	1.579830E-04	.90 QUANTILE		8.961C0E-01
FARM MEAN	7.756207E-01					MAXIMUM		1.000000E00

Figure 5

CASE 2 LOG-GAMMA 80 PERCENT LCLS WITH MODIFICATION THREE



CELL WIDTH = 5.000000E-02

CENTRAL TENDENCY		SPREAD		HIGHER CENTRAL MOMENTS		DISTRIBUTION		
MEAN	8.053215E-01	VARIANCE	3.571163E-03	M3	4.337372E-05	MINIMUM		6.585300E-01
MEDIAN	8.065500E-01	STD DEV	5.975921E-02	M4	3.486217E-05	.10 QUANTILE		7.305000E-01
TRIMEAN	8.065750E-01	COEF VAR	7.383866E-02	SKEWNESS	2.032648E-01	.25 QUANTILE	(HINGE)	7.677200E-01
MIDMEAN	8.046906E-01	MEAN DEV	4.633262E-02	KURTOSIS	-2.663959E-01	.50 QUANTILE	(MEDIAN)	8.065500E-01
MIDRANGE	8.165050E-01	RANGE	3.167500E-01	BETA1	4.311879E-05	.75 QUANTILE	(HINGE)	8.470000E-01
GEOM MEAN	8.071282E-01	MIDSPREAD	7.926000E-02	BETA2	3.473585E-05	.90 QUANTILE		9.063300E-01
HARM MEAN	8.045418E-01					MAXIMUM		9.752800E-01

APPENDIX A

APPROXIMATION OF S BY A TAYLOR SERIES

The Taylor series approximation of $f(x)$ about $x = a$ can be stated as:

$$f(x) = f(a) + f'(a)(x-a) + \frac{f''(a)(x-a)^2}{2!} + \ldots + \ldots + \frac{f^i(a)(x-a)^i}{i!} + \ldots \quad (A.1)$$

Here, $f(x) = -\ln(x)$, where $x = (1 - Q_i)$. Let $a = 1$.

Then $(x - a) = -Q_i$

$$f(a) = 0$$

$$f'(x) = \frac{1}{1 - Q_i}, \quad f'(a) = -1$$

$$f''(x) = \frac{1}{(1 - Q_i)^2}, \quad f''(a) = 1$$

$$f'''(x) = \frac{-2}{(1 - Q_i)^3}, \quad f'''(a) = -2$$

Equation (A.1) becomes:

$$f(x) = 0 + (-1)(-Q_i) + \frac{(1)(-Q_i)^2}{2!} + \frac{(-2)(-Q_i)^3}{3!} + \ldots$$

$$= Q_i + \frac{Q_i^2}{2} + \frac{Q_i^3}{3} + \ldots\ldots\ldots + \frac{Q_i^j}{j} + \ldots\ldots$$

$$= \sum_{j=1}^{\infty} \frac{Q_i^j}{j} \quad (A.2)$$

Thus, $$S = \sum_{i=1}^{K} -\ln(1 - Q_i) = \sum_{i=1}^{K} \sum_{j=1}^{\infty} \frac{Q_i^j}{j} \quad (A.3)$$

Dropping terms of order three and higher,

$$S \approx \sum_{i=1}^{K} \left[Q_i + \frac{Q_i^2}{2} \right] \quad (A.4)$$

APPENDIX B

AN UNBIASED ESTIMATOR FOR T_i

Let $T_i = Q_i + \frac{Q_i^2}{2}$. An unbiased estimator $\hat{T}_i$ is desired for T_i, which is of the form

$$\hat{T}_i = A_i \hat{Q}_i + \frac{B_i}{2} \hat{Q}_i^2 \tag{B.1}$$

where $\hat{Q}_i = \frac{F_i}{N_i}$. The number of failures of ith component F_i is a binomial ($N = N_i$, $p = Q_i$) random variable with mean $N_i Q_i$ and variance $N_i Q_i (1 - Q_i)$. Thus,

$$E(\hat{Q}_i) = \frac{1}{N_i} E(F_i) = Q_i \tag{B.2}$$

$$Var(\hat{Q}_i) = \frac{1}{N_i^2} Var(F_i) = \frac{Q_i(1 - Q_i)}{N_i} \tag{B.3}$$

$$E(\hat{Q}_i^2) = Var(\hat{Q}_i) + (E(\hat{Q}_i))^2 \tag{B.4}$$

$$= \frac{Q_i(1 - Q_i)}{N_i} + Q_i^2$$

$$= \frac{Q_i - Q_i^2 + N_i Q_i^2}{N_i}$$

Substituting

$$E(\hat{T}_i) = A_i E(\hat{Q}_i) + \frac{B_i}{2} E(\hat{Q}_i^2)$$

$$= (A_i + \frac{B_i}{2N_i}) Q_i + (B_i - \frac{B_i}{N_i}) \frac{Q_i^2}{2} \tag{B.5}$$

That $\hat{T}_i$ is unbiased implies

$$E(\hat{T}_i) = T_i = Q_i + \frac{Q_i^2}{2} \tag{B.6}$$

Equating coefficients with equation (B.5)

$$A_i + \frac{B_i}{2N_i} = 1 \tag{B.7}$$

$$B_i - \frac{B_i}{N_i} = 1 \tag{B.8}$$

Solving the above two equations simultaneously gives the desired expressions for A_i and B_i

$$A_i = \frac{2N_i - 3}{2N_i - 2} \tag{B.9}$$

$$B_i = \frac{N_i}{N_i - 1} \tag{B.10}$$

APPENDIX C

DERIVATION OF THE ACTUAL VARIANCE OF $\hat{T}_i$ AND COMPARISON WITH $\sum_{i=1}^{K} \frac{\hat{T}_i}{N_i}$

Reference 1 used $\frac{\hat{T}_i}{N_i}$ as a variance estimate for $\hat{T}_i$. Then for the method of moments gamma fit,

$$\mathrm{Var}(\hat{S}) = \sum_{i=1}^{K} \mathrm{Var}(\hat{T}_i) \approx \sum_{i=1}^{K} \frac{\hat{T}_i}{N_i} \tag{C.1}$$

The following is a derivation of the actual variance of $\hat{T}_i$, and then a comparison with the above estimator.

Since $$\hat{T}_i = A_i \hat{Q}_i + \frac{B_i}{2} \hat{Q}_i^2 \tag{C.2}$$

then $$\mathrm{Var}(\hat{T}_i) = A_i^2 \mathrm{Var}(\hat{Q}_i) + A_i B_i \mathrm{COV}(\hat{Q}_i, \hat{Q}_i^2) + \frac{B_i^2}{4} \mathrm{Var}(\hat{Q}_i^2) \tag{C.3}$$

where $\hat{Q}_i = \frac{F_i}{N_i}$. The total number of failures on the ith component F_i, is a binomial ($N = N_i$, $p = Q_i$) random variable. Then

$$E(\hat{Q}_i) = \frac{1}{N_i} E(F_i) = Q_i \tag{C.4}$$

$$\mathrm{Var}(\hat{Q}_i) = \frac{1}{N_i^2} \mathrm{Var}(F_i) = \frac{Q_i(1 - Q_i)}{N_i} \tag{C.5}$$

$$E(\hat{Q}_i^2) = \mathrm{Var}(\hat{Q}_i) + (E(\hat{Q}_i))^2$$

$$= \frac{Q_i - Q_i^2}{N_i} + Q_i^2 \tag{C.6}$$

and $$\mathrm{COV}(\hat{Q}_i, \hat{Q}_i^2) = E(\hat{Q}_i^3) - E(\hat{Q}_i) E(\hat{Q}_i^2) \tag{C.7}$$

To solve equation (C.7), $E(\hat{Q}_i^3) = \frac{1}{N_i^3} E[F_i^3]$ must be solved.

Model $I_m = \begin{cases} 1 & \text{if the } m^{th} \text{ test fails} \\ 0 & \text{if the } m^{th} \text{ test does not fail} \end{cases}$

and let $F_i = \sum_{m=1}^{N_i} I_m$ (C.8)

Then $F_i^3 = \left(\sum_{m=1}^{N_i} I_m \right)^3 = \sum_j \sum_k \sum_l I_j I_k I_l$ (C.9)

and $E(F_i^3) = \underset{\text{(all three equal)}}{\sum_{j=k=l}} E(I_j^3) + \underset{\text{(two the same and one distinct)}}{\Sigma\Sigma} E(I_j^2)E(I_k) +$

$$\underset{\text{(all three distinct)}}{\sum_j \sum_k \sum_l} E(I_j)E(I_k)E(I_l) \qquad \text{(C.10)}$$

$$= N_i Q_i + 3N_i(N_i - 1)Q_i^2 + N_i(N_i - 1)(N_i - 2)Q_i^3$$

This implies $E(\hat{Q}_i^3) = \frac{1}{N_i^3} E(F_i^3)$

$$= \frac{1}{N_i^2} \left[Q_i + 3(N_i - 1)Q_i^2 + (N_i - 1)(N_i - 2)Q_i^3 \right]$$

Now, using equation (C.7),

$$COV(\hat{Q}_i, \hat{Q}_i^2) = \frac{Q_i}{N_i^2} \left\{ 1 + (2N_i - 3)Q_i - 2(N_i - 1)Q_i^2 \right\} \qquad \text{(C.11)}$$

All that remains unknown in equation (C.3) is $Var(\hat{Q}_i^2)$.

$$Var(\hat{Q}_i^2) = E(\hat{Q}_i^4) - (E(\hat{Q}_i^2))^2$$

$$= \frac{1}{N_i^4} E(F_i^4) - \left\{ \frac{Q_i - Q_i^2}{N_i} + Q_i^2 \right\}^2 \qquad \text{(C.12)}$$

Using the same model as before,

$$E(F_i^4) = \sum_{j=k=l=mn} E(I_j^4) + \sum_j \sum_k E(I_j^2) + \sum_j \sum_k E(I_j^3) E(I_k)$$

(all indexes equal) (indexes equal in pairs) (three indexes the same and one distinct)

$$+ \sum_j \sum_k \sum_l E(I_j^2) E(I_k) E(I_l)$$

(two indexes the same and two distinct)

$$+ \sum_j \sum_k \sum_l \sum_{mn} E(I_j) E(I_k) E(I_l) E(I_{mn}) \qquad (C.13)$$

(all four indexes distinct)

$$= N_i Q_i + 3N_i(N_i - 1)Q_i^2 + 4N_i(N_i - 1)Q_i^2 + 6N_i(N_i - 1)(N_i - 2)Q_i^3$$

$$+ N_i(N_i - 1)(N_i - 2)(N_i - 3)Q_i^4$$

Equation (C.13) is used to find

$$E(\hat{Q}_i^4) = \frac{1}{N_i^3}\left[Q_i + 7(N_i - 1)Q_i^2 + 6(N_i - 1)(N_i - 2)Q_i^3 + (N_i - 1)(N_i - 2)(N_i - 3)Q_i^4\right] \qquad (C.14)$$

Substituting equation (C.14) into equation (C.12),

$$\mathrm{Var}(\hat{Q}_i^2) = \frac{Q_i}{N_i^3}\left\{1 + (6N_i - 7)Q_i + 4(N_i - 1)(N_i - 3)Q_i^2 - 2(N_i - 1)(2N_i - 3)Q_i^3\right\} \qquad (C.15)$$

The final result can now be obtained by substituting equations (C.15), (C.11), and (C.5) into equation (C.3), yielding

$$\mathrm{Var}(\hat{T}_i) = \frac{A_i^2}{N_i}(Q_i - Q_i^2) + A_i B_i \frac{Q_i}{N_i^2}\left[1 + (2N_i - 3)Q_i - 2(N_i - 1)Q_i^2\right]$$

$$+ \frac{B_i^2}{4}\frac{Q_i}{N_i^3}\left\{1 + (6N_i - 7)Q_i + \left[4(N_i - 1)(N_i - 3)\right]Q_i^2\right.$$

$$\left. - 2(N_i - 1)(2N_i - 3)Q_i^3\right\}^2 \qquad \text{(C.16)}$$

The cumbersome expression above can be greatly simplified by dropping terms with divisors of N_i^2 and higher, yielding

$$\mathrm{Var}(\hat{T}_i) \approx \frac{Q_i}{N_i}\left[(1 - Q_i)(A_i + B_i Q_i)^2\right] \qquad \text{(C.17)}$$

Then $\widehat{\mathrm{Var}}(\hat{T}_i) = \frac{\hat{Q}_i}{N_i}\left[(1 - \hat{Q}_i)(A_i + B_i\hat{Q}_i)^2\right]$ where $\hat{Q}_i = \frac{F_i}{N_i}$ (C.18)

A comparison was made between $\widehat{\mathrm{Var}}(\hat{T}_i)$ as given by equation (C.18), and that used by Ref. 1. The results are listed for $N_1 = 10$, 20, and 50 in Table XV. The simplistic estimator $\sum_{i=1}^{K} \frac{\hat{T}_i}{N_i}$ turns out to be an excellent approximation for $\mathrm{Var}(\hat{S}) = \sum_{i=1}^{K} \mathrm{Var}(\hat{T}_i)$.

TABLE XV

Actual Variance of $\hat{T}_i$ Compared to Estimate

(One Failure Per Component)

FAILURES	$\widehat{Var}(\hat{S}) = \sum_{i=1}^{K} \frac{\hat{T}_i}{N_i}$	$\widehat{Var}(\hat{S}) = \sum_{i=1}^{K} \frac{\hat{Q}_i}{N_i}(1-\hat{Q}_i)(A_i+B_i\hat{Q}_i)^2$
1 N_1=10	.01000	.01003
20	.00250	.00250
50	.00040	.00040
2 N_1=10	.02235	.02242
20	.Q0527	.00527
50	.00082	.00082
3 N_1=10	.03469	.03480
20	.00804	.00805
50	.00123	.00123
4 N_1=10	.05032	.05050
20	.01113	.01113
50	.00167	.00167
5 N_1=10	.06594	.06619
20	.01421	.01422
50	.00210	.00210
6 N_1=10	.08635	.08672
20	.01767	.01769
50	.00255	.00255
7 N_1=10	.10676	.10725
20	.02113	.02115
50	.00301	.00301
8 N_1=10	.13454	.13526
20	.02504	.02506
50	.00348	.00348
9 N_1=10	.16231	.16327
20	.02895	.02897
50	.00395	.00395
10 N_1=10	.20231	.20377
20	.03339	.03342
50	.00445	.00445

APPENDIX D

DERIVATION OF THE ZERO FAILURE CASE FOR THE MODIFIED LIEBERMAN-ROSS METHOD

The variables U and I are undefined when no failures occur. Consider type III censoring in which N components are placed on simultaneous test, with replacement, until a pre-selected time T_0 [Ref. 8, p. 238]. Type III censoring is a component LCL procedure, and not a system LCL procedure. Now, consider the Modified Lieberman-Ross (MLR) procedure where N_i items of the ith component have been tested with no failures, $i = 1,2,\ldots,K$. If the N_i are identical (as is the case in this study for zero failures), then the above is equivalent to testing N_i "systems" with no failures until $T_0 = 1$ "system" mission.

The type III 100(1-α)% LCL on the mission probability of success p of a component is

$$\hat{p}_{L(\alpha)} = \text{Exp}\left(\frac{-\chi^2_{\alpha,2F+2}}{2NT_0}\right) \qquad (D.1)$$

where F is the observed number of failures, N is the number of components on simultaneous test, and T_0 is the preselected test time. Then for $F = 0$, $2F + 2 = 2$ and

$$2NT_0 = 2N_i \cdot 1 \text{ system mission unit} = 2N_i$$

A 100(1-α)% LCL on system reliability R_s when no failures occur is

$$\hat{R}_{S,L(\alpha)} = \text{Exp}\left(\frac{-\chi^2_{\alpha,2}}{2N_i}\right) \qquad (D.2)$$

for N_i equal, $i=1,2,\ldots,K$. If the N_i are not equal, the minimum N_i should be used.

APPENDIX E

USE OF THE PROBABILITY INTEGRAL TRANSFORM (PIT) IN GENERATING EXPONENTIAL AND WEIBULL FAILURE DATA

Assume an exponential failure process for the ith component with rate λ_i per mission unit. Then the reliability of the ith component for completing one mission unit is:

$$R_i(1) = \text{Exp}(-\lambda_i \cdot 1) \qquad \text{(F.1)}$$

Utilizing the PIT method, draw a uniform (0,1) random number, denoted "U", and form the equality:

$$U = \text{Exp}(-\lambda_i t) \qquad \text{(F.2)}$$

Solving for time-to-failure t

$$\ln(U) = -\lambda_i t \qquad \text{(F.3)}$$

It follows that:

$$t = \frac{-\ln(U)}{\lambda_i} \qquad \text{(F.4)}$$

The computer program uses the following algorithm:

1) Generate a uniform (0,1) random variable

2) If $U = \text{Exp}(-\lambda_i t)$ is less than $\text{Exp}(-\lambda_i \cdot 1)$, then $t>1$ mission unit, and a success has occurred. Truncate the time-on-test for that component trial to 1.

3) Otherwise, a failure has occurred at $t = \frac{-\ln(U)}{\lambda_i}$

The simulation was also ran under a Weibull failure process for sensitivity analysis. The Weibull (β,λ) density function is $f_T(t) = \beta\lambda t^{\beta-1} \text{Exp}(-\lambda t^{\beta})$ (F.5)

and hazard function

$$h(t) = \beta\lambda t^{\beta-1} \qquad t \geq 0, \lambda > 0, \beta > 0 \qquad \text{(F.6)}$$

Using the PIT method as before, draw a uniform (0,1) random number U and set

$$U = \bar{F}(t)$$

$$= \text{Exp}(-\lambda_i t^{\beta}) \qquad \text{(F.7)}$$

thus $$T = \left(-\frac{\ln(U)}{\lambda_i}\right)^{1/\beta} \qquad \text{(F.8)}$$

Comparing equations (F.4) and (F.8) shows that if $T = \frac{-\ln(U)}{\lambda_i}$ is an exponential (λ_i) component failure time, then $T^{1/\beta}$ is Weibull (β, λ_i). Thus, to perform the Weibull sensitivity analysis, only the above change needs to be made to the simulation algorithm (Chapter III).

Taking the derivative of the Weibull hazard function with respect to time

$$\frac{d(h(t))}{dt} = \lambda_i(\beta^2 - \beta)t^{\beta-2} \qquad \text{(F.9)}$$

Therefore, any $0 < \beta < 1$ yields a decreasing hazard rate, and any $\beta > 1$ gives an increasing one. Case one was ran with $\beta = .5$ and 3.0. Note that the comparison logic, used by the simulation algorithm to determine if a failure has occurred, reduces to the original expression. The number of failures is then the same as before. The Weibull results are shown in Table IV.

APPENDIX F

DERIVATION OF THE RANDOMIZED LOG-GAMMA METHOD

The technique for randomization of the classical binomial model is a direct application of Barr and Jayachandran's 1975 article [Ref. 7]. It can be applied directly to obtain a system reliability LCL when a total of zero or one failure occurs among all component tests, because these cases translate into equivalent system failures (see Figure 3, p. 45).

The binomial point estimate for p is $\hat{p} = X/N$, where X is the number of observed successes in N trials. The classical binomial $100(1-\alpha)\%$ LCL for p, denoted $\hat{p}_{L(\alpha)}$, is the solution for p in the equation

$$\sum_{j=X}^{N} \binom{N}{j} p^j (1-p)^{N-j} = \alpha \qquad (F.1)$$

when no failures occur, or $X = N$, this becomes

$$\hat{p}_{L(\alpha)} = \alpha^{1/N} \qquad (F.2)$$

and for one failure, $\hat{p}_{L(\alpha)}$ is the solution for p in

$$Np^{N-1}(1-p) + p^N = \alpha \qquad (F.3)$$

This $\hat{p}_{L(\alpha)}$ has the property that $\Pr\left[\hat{p}_{L(\alpha)} \leq p\right] \geq 1 - \alpha$

Reference 7 develops a method of randomizing discrete LCL procedures which are conservative, and illustrates it for the binomial case. The exact randomized $100(1-\alpha)\%$ binomial LCL for p is developed as follows.

a. The number of observed successes X is a binomial random variable with probability mass function $B(x;N,p)$. Define $h(p)$ as the smallest integer such that

$$\Pr(X > h(p)) \leq \alpha \qquad x \ 0,1,\ldots,N \tag{F.4}$$

Thus, $h(p)$ denotes an observed value of X.

b. Define

$$\alpha_p \equiv \Pr(X > h(p)) \tag{F.5}$$

and

$$y(p) \equiv F^{-1}\left\{1 - (\alpha - \alpha_p)/B(h(p);N,p)\right\} \tag{F.6}$$

where F is the "randomizing" distribution function of Y. Reference 7 shows that F can be the uniform (0,1) distribution function without loss of generality. Then using equation (F.5), $y(p)$ can be simplified to

$$y(p) = 1 - (\alpha - \Pr(X > h(p))/\Pr(X=h(p)) \tag{F.7}$$

Finally, define

$$g(p) \equiv h(p) + y(p) \tag{F.8}$$

Then the exact binomial randomized LCL for p, $L^*(x+y)$ is given by

$$L^*(x+y) = \begin{cases} g^{-1}(x+y); & 1-\alpha \leq x+y \leq N+1-\alpha \\ 0 \quad ; & x+y < 1-\alpha \\ 1.0 \quad ; & x+y > N+1-\alpha \end{cases} \tag{F.9}$$

To solve for $L^*(x+y)$, form the equality

$$g(p) = X + Y = h(p) + y(p) \tag{F.10}$$

Since $h(p)$ is an observed value of X, this reduces to

$$Y = y(p) = 1 - (\alpha - \Pr(X>h(p)))/\Pr(X = h(p)) \tag{F.11}$$

By solving for p and applying equation (F.9)

$$L^*(x+y) = \begin{cases} \text{the solution for p in} \\ (1-Y)\Pr(X=x) + \Pr(X>x) = \alpha & 1-\alpha \le x+y \le N+1-\alpha \\ 0 & x+y < 1-\alpha \\ 1.0 & x+y > N+1-\alpha \end{cases} \tag{F.12}$$

Thus, for no failures, or $X = N$, $L^*(x+y)$ is the solution for p in

$$(1-Y)p^N = \alpha \tag{F.13}$$

or

$$\hat{p}_{L(\alpha)} = \begin{cases} \left(\frac{\alpha}{1-Y}\right)^{1/N} & y \le 1-\alpha \\ 1.0 & y > 1-\alpha \end{cases} \tag{F.14}$$

when $X = N-1$, or one failure, $L^*(x+y)$ is the solution for p in

$$(1-Y)\Pr(X=x) + \Pr(X>x) = \alpha$$

or

$$(1-Y)Np^{N-1}(1-p) + p^N = \alpha \tag{F.15}$$

When two or more failures occur, a component LCL procedure such as $L^*(x+y)$ cannot be used. The original Log-Gamma procedure can, however, be "randomized" as follows. Draw a uniform (0,1) random number for each type of component that fails. Denote it Y_i for the ith component, and compute the "randomized" number of failures

$$F^* = F_i - Y_i \tag{F.16}$$

where F_i is the observed number of failures on the ith component.

The Randomized Log-Gamma procedure (RLG) was developed by trial and error via computer simulation, using the above adjustments where most effective. It can be stated as follows:

Randomized Log-Gamma Method

a. Zero failures in all components:

$$\hat{R}_{S,L(\alpha)} = \begin{cases} \left(\frac{\alpha}{1-Y}\right)^{1/N_1} & y \leq 1-\alpha \\ 1.0 & y > 1-\alpha \end{cases} \qquad \text{(F.17)}$$

where $Y \backsim$ uniform (0,1).

b. One failure among all components:

$$\hat{R}_{S,L(\alpha)} = \text{the solution for p in } (1-Y)N_1 p^{N_1-1}(1-p) + p^{N_1} = \alpha \qquad \text{(F.18)}$$

The use of α^{1/N_1} as an approximation to $\hat{R}_{S,L(\alpha)}$, for $y > 1-\alpha$, is reasonably accurate for low N_1 and very accurate for large N_1, as shown below.

TABLE XVI

Accuracy of the Approximation α^{1/N_1}

$Y = .9$

$\alpha = .2$

Actual $\hat{R}_{S,L(\alpha)}$ / α^{1/N_1}	N = 10	20	50
	.837	.915	.965
	.851	.923	.968

c. Two through four component failures:

For each different type of component that had at least one failure, draw a uniform (0,1) random number.

Denote it Y_i for the ith component, and compute

$$F_i^* = F_i - Y_i \qquad (F.19)$$

Compute Log-Gamma method with exact degrees of freedom and F_i^* instead of the observed number of failures F_i.

d. Five or more failures:

Use the Log-Gamma method with exact degrees of freedom.

APPENDIX G

AN EXAMPLE OF THE RANDOMIZED LOG-GAMMA PROCEDURE

Let $N_1 = 20$, $K = 15$, and $\alpha = .2$. Suppose that the replica has no failures. Then a uniform (0,1) random number Y is generated; say $Y = .5$. Then since $Y < 1 - \alpha$,

$$\hat{R}_{S,L(.2)} = \left(\frac{.2}{1 - .5}\right)^{\frac{1}{20}} = .955$$

as opposed to the classical binomial LCL of .923.

Suppose some future replica had one failure in some component, and $Y = .6$ is generated. Then $\hat{R}_{S,L(.2)}$ is the solution for p in the equation

$$20p^{19}(1 - p)(1 - .6) + p^{20} = .2$$

Therefore, $\hat{R}_{S,L(.2)} = .892$. The classical binomial LCL is .858.

Now, suppose another replica had two failures in the 5th component and one failure in the 15th component, and that $Y_5 = .3$ and $Y_{15} = .8$ were generated. Thus, $F_5^* = 1.7$ and $F_{15}^* = .2$. Applying the LG procedure, $N_5 = 20$, $N_{15} = 19$

$$\hat{S} = \sum_{i=1}^{15} \hat{T}_i = \hat{T}_5 + \hat{T}_{15} = .0969$$

$$\hat{r} = \frac{(\hat{S})^2}{\sum_{i=1}^{15} \frac{\hat{T}_i}{N_i}} = \frac{(.0969)^2}{.0049} = 1.926$$

$$2\hat{r} = 3.853$$

$$\chi^2_{.8,3.853} \approx 1.56$$

Then $$\hat{R}_{S,L(.2)} = \text{Exp}\left\{ \frac{-2\hat{r}\hat{S}}{\chi^2_{.8,3.853}} \right\} = .787$$

without randomization, the LCL would have been

$$\hat{R}_{S,L(.2)} = .740$$

APPENDIX H

THE COMPUTER PROGRAM

The following pages list the computer program used for the simulations. Flags were used to designate the portions of the program to be executed, depending on which modifications were desired. Proprietary subroutines [Ref. 9] were utilized for inverse chi square values (MDCHI), sorting the approximate distributions of $\hat{R}_{S,L(\alpha)}$, (VSORTA), generation of uniform (0,1) variates (GGUB), and solving for the RLG procedure (ZREAL2). The APL subroutines HISTLIST and HIST were used in Table X and Figures 4 and 5 respectively [Ref. 10] . The plotting subroutine PLOTP was used in Tables VII and VIII [Ref. 11].

The program was written in the FORTRAN IV language for use on the IBM-360 system at the Naval Postgraduate School.

```
      COMMON REALNI,UUU,ALPHA                                          TTT00010
      DIMENSION RLA(500),RL9(500),RDB(500),RD9(500),RTB(500),RT9(500)  TTT00020
      DIMENSION NF(40),KF(40,250),RAMBDA(40),T(40,250)                 TTT00030
      DIMENSION A(40),ZE(40),ZT(40),KFSUM(40),NN(40),TSUM(40),B(40)    TTT00040
      DIMENSION TLAST(800),KFLAST(800),TLLAST(800)                     TTT00050
      DIMENSION L(40,250),Z(800)                                       TTT00060
      DIMENSION TLSUM(40),TL(40,250),TLROSS(40)                        TTT00070
      DIMENSION PNEWT(2),PNEWT3(3)                                     TTT00080
      EXTERNAL FCN                                                     TTT00090
      FINAL=1.                                                         TTT00100
      BETA=1.                                                          TTT00110
      N1=150                                                           TTT00120
      NUSE=N1                                                          TTT00130
      NK=13                                                            TTT00140
      NKM1=NK-1                                                        TTT00150
      DO 9090 K=1,NKM1                                                 TTT00160
 9090 R(K)=.997                                                        TTT00170
      R(1)=.997                                                        TTT00180
      R(NK)=.997                                                       TTT00190
      NN(1)=150                                                        TTT00200
      NN(2)=30                                                         TTT00210
      NN(3)=75                                                         TTT00220
      NN(4)=100                                                        TTT00230
      NN(5)=125                                                        TTT00240
      NN(6)=18                                                         TTT00250
      NN(7)=28                                                         TTT00260
      NN(8)=125                                                        TTT00270
      NN(9)=63                                                         TTT00280
      NN(10)=125                                                       TTT00290
      NN(11)=59                                                        TTT00300
      NN(12)=5                                                         TTT00310
      NN(13)=19                                                        TTT00320
C     NN(14)=30                                                        TTT00330
      R(1)=.995                                                        TTT00340
      R(2)=.985                                                        TTT00350
      R(3)=.979                                                        TTT00360
      R(4)=.988                                                        TTT00370
      R(5)=.982                                                        TTT00380
      R(6)=.98                                                         TTT00390
      R(7)=.967                                                        TTT00400
      R(8)=.995                                                        TTT00410
      R(9)=.97                                                         TTT00420
      R(10)=.995                                                       TTT00430
      R(11)=.968                                                       TTT00440
      R(12)=.98                                                        TTT00450
      R(13)=.9                                                         TTT00460
C     R(14)=.99                                                        TTT00470
      WRITE(6,1013)                                                    TTT00480
      WRITE(6,1000)                                                    TTT00490
      WRITE(6,101) N1,NK                                               TTT00500
      WRITE(6,1007) BETA                                               TTT00510
      WRITE(6,1009) FINAL                                              TTT00520
      WRITE(6,223) (K,R(K),K=1,NK)                                     TTT00530
C     WRITE(6,102) (IJK,NN(IJK),IJK=1,NK)                              TTT00540
      SYSREL=1.0                                                       TTT00550
      DO 55 K=1,NK                                                     TTT00560
      RAMBDA(K)=-1./ALOG(R(K))                                         TTT00570
      SYSREL=SYSREL*R(K)                                               TTT00580
   55 CONTINUE                                                         TTT00590
      KKKMAX=0                                                         TTT00600
      KKKMIN=(INT(FINAL))*N1                                           TTT00610
      MX=777777                                                        TTT00620
      KFAIL=0                                                          TTT00630
      DO 77 L=1,500                                                    TTT00640
      ZDI=0.                                                           TTT00650
      ZTI=0.                                                           TTT00660
      CU=FINAL*FLOAT(N1)                                               TTT00670
      ZDVAR=0.                                                         TTT00680
      ZTVAR=0.                                                         TTT00690
      KTOTAL=0                                                         TTT00700
```

```
      NN(I )=N1                                                          TTTC0710
C  INPUTS FOR ZREAL2                                                    TTTC0720
      EPS=1.E-03                                                         TTTC0730
      EPS2=1.E-03                                                        TTTC0740
      ETA=1.E-03                                                         TTTC0750
      NSIG=2                                                             TTTC0760
C     NFLAG 0 MEANS N1 EVEN                                            TTTC0770
      NFLAG=0                                                            TTTC0780
      IF(MOD(N1SE,2).NE.0) NFLAG=1                                       TTTC0790
      NRCOT=2                                                            TTTC0800
      NRCOT3=3                                                           TTTC0810
      NROUT1=1                                                           TTTC0820
      ITMAX=500                                                          TTTC0830
      DO 88 K=1,NK                                                       TTTC0840
      NTEMP=NN(K )                                                       TTTC0850
      CALL GGUP(MX,NTEMP,Z)                                              TTTC0560
      IF(K.EQ.NK) GO TO 58                                               TTTC0870
      DO 44 J=1,NTEMP                                                    TTTC0880
   44 U(K ,J)=Z(J)                                                       TTTC0890
   58 KLIEB=1                                                            TTTC0900
      KCUNT=0                                                            TTTC0910
      TLSUM(K )=0.                                                       TTTC0920
      TSUM(K )=0.                                                        TTTC0930
      KFSUM(K )=0                                                        TTTC0940
      DO 99 J=1,NTEMP                                                    TTTC0950
      IF(K.EQ.NK) GO TO 27                                               TTTC0960
      IF((U(K ,J).LE.1.).AND.(U(K ,J).GE.R(K ))) GO TO 5                 TTTC0970
      GO TO 18                                                           TTTC0980
   27 IF ((Z(J).LE.1.).AND.(Z(J).GE.R(K))) GO TO 57                      TTTC0990
      TLAST(J)=1.                                                        TTTC1000
      KFLAST(J)=0                                                        TTTC1010
      GO TO 58                                                           TTTC1020
   18 KF(K ,J)=0                                                         TTTC1030
      T(K ,J)=1.                                                         TTTC1040
C  FAILURE                                                              TTTC1050
      GO TO 10                                                           TTTC1060
   57 KFLAST(J)=1                                                        TTTC1070
      KCUNT=KCUNT+1                                                      TTTC1080
      TLAST(J)=((-1.*ALOG(Z(J)))/RAMBDA(K))**(1./BETA)                   TTTC1090
      TLLAST(KCUNT)=TLAST(J)*FLOAT(J-KLIEB)                              TTTC1100
      KLIEB=J+1                                                          TTTC1110
      TLSUM(K)=TLSUM(K)+TLLAST(KCUNT)                                    TTTC1120
      GO TO 58                                                           TTTC1130
    5 KF(K ,J)=1                                                         TTTC1140
      KCUNT=KCUNT+1                                                      TTTC1150
      T(K ,J)=((-1.*ALOG(U(K ,J)))/RAMBDA(K))**(1./BETA)                 TTTC1160
      TL(K ,KCUNT)=T(K,J)*FLOAT(J-KLIEB)                                 TTTC1170
      KLIEB=J+1                                                          TTTC1180
      TLSUM(K )=TLSUM(K )+TL(K ,KOUNT)                                   TTTC1190
      GO TO 10                                                           TTTC1200
   58 TSUM(K)=TSUM(K)+TLAST(J)                                           TTTC1210
      KFSUM(K)=KFSUM(K)+KFLAST(J)                                        TTTC1220
      GO TO 99                                                           TTTC1230
   10 TSUM(K )=TSUM(K )+T(K ,J)                                          TTTC1240
      KFSUM(K )=KFSUM(K )+KF(K ,J)                                       TTTC1250
   99 CONTINUE                                                           TTTC1260
      KTOTAL=KTOTAL+KFSUM(K)                                             TTTC1270
      IF(KFSUM(K).EQ.0) GO TO 65                                         TTTC1280
      IF(TLSUM(K).LT.CC) CC=TLSUM(K)                                     TTTC1290
   65 CONTINUE                                                           TTTC1300
      IF(KCUNT.EQ.0)TLSUM(K)=0                                           TTTC1310
      A(K )=(2.*FLOAT(NTEMP)-3.)/(2.*FLOAT(NTEMP)-2.)                    TTTC1320
      B(K )=FLOAT(NTEMP)/(FLOAT(NTEMP)-1.)                              TTTC1330
      RKFSUM=FLOAT(KFSUM(K))                                             TTTC1340
      ZD(K)=A(K)*(RKFSUM/FLOAT(NTEMP))+(B(K)/2.)*(RKFSUM/FLOAT(NTEMP))** TTTC1350
     1*2                                                                 TTTC1360
      ZDVAR=ZDVAR+ZD(K)/FLOAT(NN(K))                                     TTTC1370
      ZDT=ZDT+ZD(K)                                                      TTTC1380
      ZT(K )=FLOAT(KFSUM(K ))/TSUM(K )                                   TTTC1390
      ZTVAR=ZTVAR+ZT(K)/FLOAT(NN(K))                                     TTTC1400
```

```
      ZTT=ZTT+ZT(K)                                                     TTT01410
      IF(K.EQ.NK) GO TO 88                                              TTTC1420
      CCUNT=FLOAT(NT)-.5*FLOAT(KTOTAL)                                  TTTC1430
      IF(KCUNT.GT.0) NTEMP=INT(CCUNT)                                   TTT01440
      KTEMP=K+1                                                         TTTC1450
C     NN(KTEMP)=NTEMP                                                   TTT01460
      IF(KTEMP.EQ.NK)GO TO 16                                           TTT01470
      GO TO 26                                                          TTT01480
   16 CCUNT=FINAL*(FLOAT(NTEMP))                                        TTTC1490
      NTEMP=INT(CCUNT)                                                  TTTC1500
C     NN(KTEMP)=NTEMP                                                   TTTC1510
   26 CONTINUE                                                          TTT01520
   88 CONTINUE                                                          TTT01530
      IF(KTOTAL.LT.KKKMIN) KKKMIN=KTOTAL                                TTT01540
C COMPUTE MAX AND MIN FAILURES                                          TTT01550
      IF(KTOTAL.GT.KKKMAX) KKKMAX=KTOTAL                                TTT01560
      IMROSS=0                                                          TTTC1570
      IF(KTOTAL.EQ.0) GO TO 70                                          TTT01580
      DO 66 KK=1,NK                                                     TTTC1590
      ILRCSS(KK)=0                                                      TTTC1600
      IF(KFSUM(KK).EQ.0) GO TO 66                                       TTTC1610
      KSAVE=KFSUM(KK)                                                   TTTC1620
      TLTEMP=0.                                                         TTTC1630
      DO 33 KKK=1,KSAVE                                                 TTTC1640
      IF(KK.EQ.NK) GO TO 90                                             TTTC1650
      TLTEMP=TLTEMP+FL(KK,KKK)                                          TTT01660
      GO TO 91                                                          TTTC1670
   90 TLTEMP=TLTEMP+FLLAST(KKK)                                         TTTC1680
   91 IF(TLTEMP.LT.UU) GO TO 66                                         TTTC1690
      ILRCSS(KK)=ILRCSS(KK)+1                                           TTTC1700
   33 CONTINUE                                                          TTT01710
   66 IMRCSS=IMRCSS+ILRCSS(KK)                                          TTTC1720
      GO TO 85                                                          TTT01730
   70 UU=FLOAT(NUSE)                                                    TTT01740
C C FAILURE MOD                                                         TTTC1750
      RKFSUM=.5                                                         TTTC1760
      ZDT=A(1)*(RKFSUM/FLOAT(BB(1)))+(B(1)/2.)*(RKFSUM/FLOAT(NN(      TTT01770
     1))**2                                                             TTTC1780
   85 CONTINUE                                                          TTT01790
      IMAX=MAXO(1,IMROSS)                                               TTTC1800
      CALL MDCHI(.8,FLOAT(2*IMAX),XL8,I5)                               TTTC1810
      CALL MDCHI(.9,FLOAT(2*IMAX),XL9,I6)                               TTT01820
      ULTEST=1.E-02                                                     TTTC1830
      IF(UU.LE.ULTEST) WRITE(6,9782) UU                                 TTTC1840
 9782 FORMAT(1X,' UU FLAG,UU=',F20.7)                                   TTTC1850
      RL8(L)=EXP(-XL8/(2.*UU))                                          TTTC1860
      RL9(L)=EXP(-XL9/(2.*UU))                                          TTT01870
      IF(KTOTAL.EQ.0) GO TO 9780                                        TTTC1880
C MOD ALLOWING EXACT DEGREES OF FREEDOM                                 TTTC1890
      XLD=ZDT**2/ZDVAR                                                  TTT01900
      XLT=ZTT**2/ZTVAR                                                  TTTC1910
      XXLD=2.*XLD                                                       TTTC1920
      XXLT=2.*XLT                                                       TTT01930
      GO TO 9781                                                        TTT01940
 9780     XXLD=2.                                                       TTT01950
      XXLT=2.                                                           TTTC1960
 9781 CONTINUE                                                          TTTC1970
C  PES  2.5 MOD                                                         TTTC1980
      XXLD9=XXLD                                                        TTTC1990
      IF(XXLD9.LT.2.5) XXLD9=2.5                                        TTTC2000
      CALL MDCHI(.2,XXLD,X18,I1)                                        TTTC2010
      CALL MDCHI(.1,XXLD9,X19,I2)                                       TTTC2020
      CALL MDCHI(.2,XXLT,X18,I3)                                        TTTC2030
      CALL MDCHI(.1,XXLT,X19,I4)                                        TTTC2040
C BIAS MOD                                                              TTTC2050
      DENOM=XD8                                                         TTTC2060
      IDENS=XD9                                                         TTTC2070
      RD8(L)=EXP((-1.*XXLD*ZDT)/DDENO)                                  TTTC2080
      RD9(L)=EXP((-1.*XXLD9*ZDT)/DDEN9)                                 TTTC2090
      RT8(L)=EXP((-1.*XXLT*ZTT)/XT3)                                    TTTC2100
```

```
      RT9(L)=EXP((-1.**XL1*ZTT)/XT9)                                    TTTC2110
C MOD M-1                                                               TTTC2120
      IMOD1=0                                                           TTTC2130
      IF ((IMOD1.NE.1) GO TO 4103                                       TTTC2140
      IF(KTOTAL.GT.0) GO TO 4100                                        TTTC2150
C  0 FAILURES                                                           TTTC2160
      RD8(L)=.8*(.2**(1./FLOAT(NL)))+.2                                 TTTC2170
      RD9(L)=.8*(.1**(1./FLOAT(NL)))+.2                                 TTTC2180
      GO TO 4102                                                        TTTC2190
 4100 IF(KTOTAL.NE.1) GO TO 4102                                        TTTC2200
C  1 FAILURE                                                            TTTC2210
      RD8(L)=.5*(.2**(1./FLOAT(NL)))+.8576)                             TTTC2220
      RD9(L)=.5*(.1**(1./FLOAT(NL)))+.8193)                             TTTC2230
      GO TO 4103                                                        TTTC2240
 4102 IF(KTOTAL.GT.8) GO TO 4103                                        TTTC2250
C  BETWEEN(AND INCL.) 2 AND 8 FAILURES                                  TTTC2260
      RD8SAV=RD8(L)                                                     TTTC2270
      RD9SAV=RD9(L)                                                     TTTC2280
C  DETERMINE LAST COMPONENT TO FAIL                                     TTTC2290
      DO 4444 IMOD=1,NK                                                 TTTC2300
      IF (NFSUM(IMOD).EQ.0) GO TO 4444                                  TTTC2310
      KLAST=IMOD                                                        TTTC2320
 4444 CONTINUE                                                          TTTC2330
      RNN=FLOAT(NN(KLAST))                                              TTTC2340
      ATEMP=(2.*RNN-3.)/(2.*RNN-2.)                                     TTTC2350
      BTEMP=RNN/(RNN-1.)                                                TTTC2360
      TTEMP=ATEMP*(1./RNN)*(BTEMP/2.)*(1./RNN)*(1./RNN)                 TTTC2370
      ZDTNEW=ZDT-TTEMP                                                  TTTC2380
      ZDVARN=ZDVAR-(TTEMP/RNN)                                          TTTC2390
      RNEW=(ZDTNEW**2)/ZDVARN                                           TTTC2400
      RNEW=2.*RNEW                                                      TTTC2410
      RNEW9=RNEW                                                        TTTC2420
      CALL MDCHI(.2,RNEW,XD8NEW,IF)                                     TTTC2430
      IF(RNEW.LT.2.5) RNEW9=2.5                                         TTTC2440
      CALL MDCHI(.1,RNEW9,XD9NEW,IB)                                    TTTC2450
      RD8NEW=EXP((-1.*RNEW*ZDTNEW)/XD8NEW)                              TTTC2460
      RD9NEW=EXP((-1.*RNEW9*ZDTNEW)/XD9NEW)                             TTTC2470
      RD9(L)=.5*(RD8NEW*RD8SAV)                                         TTTC2480
      RD9(L)=.5*(RD9NEW*RD9SAV)                                         TTTC2490
 4103 CONTINUE                                                          TTTC2500
C  MOD M-2, RANDOMIZED LIMITS                                           TTTC2510
      IMOD2=1                                                           TTTC2520
      IF(IMOD2.NE.1) GO TO 5103                                         TTTC2530
      REALN1=FLOAT(NUSE)                                                TTTC2540
      CALL GGUB(MX,1,UUU)                                               TTTC2550
 5557 FORMAT(1X,' UUU=',F7.4)                                           TTTC2560
      IF(KTOTAL.GT.0) GO TO 5100                                        TTTC2570
C  0 FAILURES                                                           TTTC2580
      RD8(L)=(.27/(1.-UUU))**(1./REALN1)                                TTTC2590
      RD9(L)=(.17/(1.-UUU))**(1./REALN1)                                TTTC2600
      IF(UUU.GE..9) RD8(L)=1.                                           TTTC2610
      IF(UUU.GE..5) RD9(L)=1.                                           TTTC2620
      GO TO 5103                                                        TTTC2630
 5100 IF (KTOTAL.NE.1) GO TO 5101                                       TTTC2640
C  1 FAILURE                                                            TTTC2650
      IF(NFLAG.EQ.0) GO TO 5150                                         TTTC2660
C   N1 ODD                                                              TTTC2670
      IFS=0                                                             TTTC2680
      IER=32                                                            TTTC2690
      RD8(L)=.2**(1./REALN1)                                            TTTC2700
      IF(UUU.LT..8) GO TO 5170                                          TTTC2710
      RD9(L)=.1**(1./REALN1)                                            TTTC2720
      IF(UUU.GE..9) GO TO 5103                                          TTTC2730
      GO TO 5171                                                        TTTC2740
 5170 PNEWT3(1)=.53                                                     TTTC2750
      PNEWT3(2)=1.1                                                     TTTC2760
      PNEWT3(3)=-1.                                                     TTTC2770
      IF(NUSE.LE.25) PNEWT3(1)=.80                                      TTTC2780
      ALPHA=.2                                                          TTTC2790
      CALL ZREAL2(FCN,EPS,EPS2,ALPHA,DSTO,BROOT3,PNEWT3,ITMAX,IER)      TTTC2800
```

```
      IF (IER.EQ.34) WRITE(6,5557) UUU                                  TTTC2810
      IF(IER.EQ.34)GO TO 5103                                           TTTC2820
 5553 FORMAT(1X,' PNEWT3:',3F8.5)                                       TTTC2830
 5160 CONTINUE                                                          TTTC2840
      DO 5161 IR=1,3                                                    TTTC2850
      IF(PNEWT3(IR).LT.0.) GO TO 5161                                   TTTC2860
      IF(PNEWT3(IR).GT.1.) GO TO 5161                                   TTTC2870
      IRSAV=IR                                                          TTTC2880
 5161 CONTINUE                                                          TTTC2890
      IF(IFS.EQ.1) GO TO 5162                                           TTTC2900
      RD8(L)=PNEWT3(IRSAV)                                              TTTC2910
      ITMAX=500                                                         TTTC2920
      ALPHA=.1                                                          TTTC2930
 5171 IFS=1                                                             TTTC2940
      CALL ZREAL2(FCN,EPS,EPS2,ETA,NSIG,NROOT3,PNEWT3,ITMAX,IER)        TTTC2950
      IF(IER.EQ.34) WRITE(6,5557) UUU                                   TTTC2960
      IF(IER.EQ.34) GO TO 5103                                          TTTC2970
      ITMAX=500                                                         TTTC2980
      GO TO 5160                                                        TTTC2990
 5162 RD9(L)=PNEWT3(IRSAV)                                              TTTC3000
      GO TO 5103                                                        TTTC3010
 5150 CONTINUE                                                          TTTC3020
C  NI EVEN, TWO POSITIVE ROOTS                                          TTTC3030
      IF(UUU.LT..0) GO TO 5808                                          TTTC3040
      RD8(L)=.2**(1./REALNI)                                            TTTC3050
      IF(UUU.GE..9) RD9(L)=.1**(1./REALNI)                              TTTC3060
      IF(UUU.LE..9) GO TO 5103                                          TTTC3070
      PNEWT1=.92                                                        TTTC3080
      ALPHA=.1                                                          TTTC3090
      ITMAX=500                                                         TTTC3100
      CALL ZREAL2(FCN,EPS,EPS2,ETA,NSIG,NROOT1,PNEWT1,ITMAX,IER)        TTTC3110
      ITMAX=500                                                         TTTC3120
      IF(IER.EQ.34) RD8(L)=.1**(1./REALNI)                              TTTC3130
      IF(IER.EQ.34) GO TO 5103                                          TTTC3140
      RD8(L)=PNEWT1                                                     TTTC3150
      ITMAX=500                                                         TTTC3160
      GO TO 5103                                                        TTTC3170
 5808     PNEWT(1)=.96                                                  TTTC3180
      IF(NUSE.LE.25) PNEWT(1)=.89                                       TTTC3190
      PNEWT(2)=1.1                                                      TTTC3200
      ALPHA=.2                                                          TTTC3210
      CALL ZREAL2(FCN,EPS,EPS2,ETA,NSIG,NROOT,PNEWT,ITMAX,IER)          TTTC3220
 5557 FORMAT(1X, 'PNEWT:',2F8.5)                                        TTTC3230
      ITMAX=500                                                         TTTC3240
      IF(IER.EQ.34) GO TO 5880                                          TTTC3250
      DO 5151 MN=1,2                                                    TTTC3260
 5151 IF(PNEWT(MN).LE.1.) MNSAV=MN                                      TTTC3270
      RD8(L)=PNEWT(MNSAV)                                               TTTC3280
      ITMAX=500                                                         TTTC3290
      ALPHA=.1                                                          TTTC3300
      CALL ZREAL2(FCN,EPS,EPS2,ETA,NSIG,NROOT,PNEWT,ITMAX,IER)          TTTC3310
      IF(IER.EQ.34) WRITE(6,5557) UUU                                   TTTC3320
      DO 5152 MN9=1,2                                                   TTTC3330
 5152 IF(PNEWT(MN9).LE.1.) MN9SAV=MN9                                   TTTC3340
      RD9(L)=PNEWT(MN9SAV)                                              TTTC3350
      GO TO 5103                                                        TTTC3360
 5880 CONTINUE                                                          TTTC3370
      IER=32                                                            TTTC3380
C   FAILED TO CONVERGE                                                  TTTC3390
      PNEWT1=.95                                                        TTTC3400
      IF(PNEWT(1).LT.1.) PNEWT1=PNEWT(1)                                TTTC3410
      IF(PNEWT(1).LT.1.) GO TO 5881                                     TTTC3420
      RD8(L)=.2**(1./REALNI)                                            TTTC3430
      RD9(L)=.1**(1./REALNI)                                            TTTC3440
      GO TO 5103                                                        TTTC3450
 5881 RD8(L)=PNEWT1                                                     TTTC3460
      ITMAX=500                                                         TTTC3470
      ALPHA=.1                                                          TTTC3480
      CALL ZREAL2(FCN,EPS,EPS2,ETA,NSIG,NROOT1,PNEWT1,ITMAX,IER)        TTTC3490
      ITMAX=500                                                         TTTC3500
```

```
      RD9(L)=ENEWT1                                                     TTTC3510
      IF(IEP.EQ.34) RD9(L)=.1**(1./REALN(L))                            TTTC3520
      GO TO 5103                                                        TTTC3530
C  2 OR MORE FAILURES                                                   TTTC3540
 5101 CONTINUE                                                          TTTC3550
      IF (KTOTAL.GT.4) GO TO 5103                                       TTTC3560
      ZDTRAN=ZDT                                                        TTTC3570
      ZDVRAN=ZDVAR                                                      TTTC3580
C  DETERMINE WHICH COMPONENTS FAILED                                    TTTC3590
      DO 5555 IRAN=1,NK                                                 TTTC3600
      IF(KFSUM(IRAN).EQ.0) GO TO 5555                                   TTTC3610
      ZDTRAN=ZDTRAN-ZD(IRAN)                                            TTTC3620
      REALN=FLOAT(NN(IRAN))                                             TTTC3630
      ZDVRAN=ZDVRAN-(ZD(IRAN)/REALN)                                    TTTC3640
      FAILR=FLOAT(KFSUM(IRAN))-0.0                                      TTTC3650
      ZDRAN=A(IRAN)*FAILR/REALN*(B(IRAN)/2.)*(FAILR/REALN)**2           TTTC3660
      ZDTRAN=ZDTRAN+ZDRAN                                               TTTC3670
      ZDVRAN=ZDVRAN+ZDRAN/REALN                                         TTTC3680
      CALL GGUB(RX,1,0.0)                                               TTTC3690
 5555 CONTINUE                                                          TTTC3700
      XLDRAN=ZDTRAN**2/ZDVRAN                                           TTTC3710
      XLDRAN=2.*XLDRAN                                                  TTTC3720
      IF(XLDRAN.LT.2.) XLDRAN=2.                                        TTTC3730
      XRANS=XLDRAN                                                      TTTC3740
      CALL MDCHI(.2,XLDRAN,XDBRAN,I9)                                   TTTC3750
      RD8(L)=EXP((-1.*XLDRAN*ZDTRAN)/XD8RAN)                            TTTC3760
      CALL MDCHI(.1,XRANS,XL9RAN,I10)                                   TTTC3770
      RD9(L)=EXP((-1.*XRANS*ZDTRAN)/XD9RAN)                             TTTC3780
 5103 CONTINUE                                                          TTTC3790
      KFAIL=KFAIL+KTOTAL                                                TTTC3800
      ITMAX=500                                                         TTTC3810
      IMAP=0                                                            TTTC3820
      IF(IMAP.NE.1) GO TO 77                                            TTTC3830
      WRITE(6,8091) L,RB9(L),RD9(L)                                     TTTC3840
 8091 FORMAT(1X,'REPLICA',I3,' RD8=',F8.5,' RD9=',F8.5)                 TTTC3850
      WRITE(6,8092) (NN(KK),KK=1,NK)                                    TTTC3860
 8092 FORMAT(1X,'TRIALS ',8X,15I4)                                      TTTC3870
      WRITE (6,8093) (KFSUM(KK),KK=1,NK)                                TTTC3880
 8093 FORMAT(1X,'COMP.FAILURES ',15I4,/)                                TTTC3890
   77 CONTINUE                                                          TTTC3900
      DO 52 M1=1,496,5                                                  TTTC3910
      M2=M1+1                                                           TTTC3920
      M3=M1+2                                                           TTTC3930
      M4=M1+3                                                           TTTC3940
      M5=M1+4                                                           TTTC3950
   52    CONTINUE                                                       TTTC3960
      CALL VSORTA (RL8,500)                                             TTTC3970
      CALL VSORTA (RL9,500)                                             TTTC3980
      CALL VSORTA (RD8,500)                                             TTTC3990
      CALL VSORTA (RD9,500)                                             TTTC4000
      CALL VSORTA (RT8,500)                                             TTTC4010
      CALL VSORTA (RT9,500)                                             TTTC4020
      WRITE(6,1001) SYSEEL                                              TTTC4030
      WRITE(7,1005)                                                     TTTC4040
      WRITE(4,1003)RL8(400),RD8(400),RT8(400),RL9(450),RD9(450),        TTTC4050
     1RT9(450)                                                          TTTC4060
      IVAR=0                                                            TTTC4070
 3555 RT8SM=0.                                                          TTTC4080
      R19SM=0.                                                          TTTC4090
      RD8SM=0.                                                          TTTC4100
      RD9SM=0.                                                          TTTC4110
      RL8SM=0.                                                          TTTC4120
      RL9SM=0.                                                          TTTC4130
      IF(IVAR.EQ.1) GO TO 3666                                          TTTC4140
      DO 2020 IJ=1,500                                                  TTTC4150
      RD8SM=RD8SM+RD8(IJ)                                               TTTC4160
      RD9SM=RD9SM+RD9(IJ)                                               TTTC4170
      RL8SM=RL8SM+RL8(IJ)                                               TTTC4180
      RL9SM=RL9SM+RL9(IJ)                                               TTTC4190
      RT8SM=RT8SM+RT8(IJ)                                               TTTC4200
```

```
      RT9SM=RT9SM+RT9(IJ)                                                TTTC4210
 2020 CONTINUE                                                           TTTC4220
      RD8BAR=RE8SM/500.                                                  TTTC4230
      RD9BAR=RD9SM/500.                                                  TTTC4240
      RT8BAR=RT8SM/500.                                                  TTTC4250
      RT9BAR=RT9SM/500.                                                  TTTC4260
      RL8BAR=RL8SM/500.                                                  TTTC4270
      RL9BAR=RL9SM/500.                                                  TTTC4280
       IVAR=1                                                            TTTC4290
      GO TO 3555                                                         TTTC4300
 3666 CONTINUE                                                           TTTC4310
C COMPLETE SAMPLE VARIANCE                                               TTTC4320
      DO 3030 MN=1,500                                                   TTTC4330
      RD8DIF=(RD8(MN)-RD8BAR)**2                                         TTTC4340
      RD9DIF=(RD9(MN)-RD9BAR)**2                                         TTTC4350
      RT8DIF=(RT8(MN)-RT8BAR)**2                                         TTTC4360
      RT9DIF=(RT9(MN)-RT9BAR)**2                                         TTTC4370
      RL8DIF=(RL8(MN)-RL8BAR)**2                                         TTTC4380
      RL9DIF=(RL9(MN)-RL9BAR)**2                                         TTTC4390
      RT8SM=RT8SM+RT8DIF                                                 TTTC4400
      RT9SM=RT9SM+RT9DIF                                                 TTTC4410
      RL8SM=RL8SM+RL8DIF                                                 TTTC4420
      RL9SM=RL9SM+RL9DIF                                                 TTTC4430
      RD8SM=RD8SM+RD8DIF                                                 TTTC4440
      RD9SM=RD9SM+RD9DIF                                                 TTTC4450
 3030 CONTINUE                                                           TTTC4460
      RD8VAR=RD8SM/499.                                                  TTTC4470
      RD9VAR=RD9SM/499.                                                  TTTC4480
      RT8VAR=RT8SM/499.                                                  TTTC4490
      RT9VAR=RT9SM/499.                                                  TTTC4500
      RL8VAR=RL8SM/499.                                                  TTTC4510
      RL9VAR=RL9SM/499.                                                  TTTC4520
      RD8S=SQRT(RD8VAR)                                                  TTTC4530
      RD9S=SQRT(RD9VAR)                                                  TTTC4540
      RT8S=SQRT(RT8VAR)                                                  TTTC4550
      RT9S=SQRT(RT9VAR)                                                  TTTC4560
      RL8S=SQRT(RL8VAR)                                                  TTTC4570
      RL9S=SQRT(RL9VAR)                                                  TTTC4580
      WRITE(6,1CC8)                                                      TTTC4590
      WRITE(6,1003)RL8BAR,RL8BAR,RT8BAR,RL9BAR,RD9BAR,RT9BAR             TTTC4600
      WRITE(6,1CC6)                                                      TTTC4610
      WRITE(6,1003) RL8VAR,RD8VAR,RT8VAR,RL9VAR,RD9VAR,RT9VAR            TTTC4620
      WRITE(6,1012)                                                      TTTC4630
      WRITE(6,1003) RL8S,RD8S,RT8S,RL9S,RD9S,RT9S                        TTTC4640
      AVENUM=FLOAT(KFAIL)/500.                                           TTTC4650
      WRITE (6,1CC4) AVENUM                                              TTTC4660
      WRITE(6,1C14) KKKMIN,KKKMAX                                        TTTC4670
  100 FORMAT(2I4)                                                        TTTC4680
  101 FORMAT(1X,'N1=',I5, '   NK=',I5)                                   TTTC4690
  222 FORMAT(15F5.3)                                                     TTTC4700
  223 FORMAT(1X,(1X,'R(',I2,')=',F10.3))                                 TTTC4710
  333 FORMAT(1X,//,' FOR FAILURE ',I4,' U(',I4,',',I4,')=',F15.8)        TTTC4720
  343 FORMAT(/,1X,' TRPL COMPONENT ',I3,' THERE ARE ',I4,' FAILURES')    TTTC4730
  444 FORMAT(1X,'TSUM=',F10.3,'KFSUM=',I4)                               TTTC4740
  555 FORMAT(1X,(1X,'LAMBDA(',I4,')=',F10.3))                            TTTC4750
  666 FORMAT(1X,//,' FOR FAILURE ',I4,' Z(',I4,')=',F15.8)               TTTC4760
  777 FORMAT(1X,'A=',F10.3,5X,'B=',F10.3)                                TTTC4770
  888 FORMAT(1X,'ZD(',I3,')=',F10.3,5X,'ZT(',I3,')=',F10.3)              TTTC4780
  889 FORMAT(//,1X,' TL(',I3,I3,')=',F12.5)                              TTTC4790
  891 FORMAT(1X,' TLSUM(',I3,')=',F10.3,' NUMBER OF FAILURES THIS COMPONTTTC4800
     1NT=',I4)                                                           TTTC4810
  900 FORMAT(1X,' REPLICA ',I4,'  ZDT=',F12.5,'  ZTT=',F12.5)            TTTC4820
  901 FORMAT(1X,'  REPLICA ',I4,'  VARIANCE FOR ZDT IS ',F12.5,  ' VARITTTC4830
     1ANCE FOR ZTT IS ',F12.5)                                           TTTC4840
  902 FORMAT(1X,/////,' REPLICA ',I4,'  OF 500',/////)                   TTTC4850
  903 FORMAT(1X,//,'  REPLICA  ',I4,'  ESTIMATE FOR LD IS  ',F12.5,5X,' TTTC4860
     1ESTIMATE FOR LT IS  ',F12.5,//)                                    TTTC4870
  904 FORMAT (1X,///,'  FOR ALL REPLICAS,TOTAL NUMBER OF FAILURES IS ',ITTTC4880
     1I5,////)                                                           TTTC4890
  905 FORMAT(1X,//,' REPLICA ',I4,' ESTIMATED ZLD=',I5,' ESTIMATED ZLT='TTTC4900
```

```
     1,I5,/////)                                                        TTT04910
 906 FORMAT(1X,///,' XD8=',F12.5,' I1=',I5,' XD9=',F12.5,' I2=',I5,' XT8TTT04920
    1=',F12.5,'I3=',I5,' XT9=',F12.5,' I4=',I5,/////)                   TTTC4930
 907 FORMAT(1X,///,'  C SERIES , RD.8(',I4,')=',F12.5,' RD.9(',I4,')=',TTT04940
    1F12.5,//////)                                                      TTT04950
 908 FORMAT(1X,///,'  T SERIES , RT.8(',I4,')=',F12.5,' RT.9(',I4,')=',TTTC4960
    1F12.5,//////)                                                      TTT04970
 909 FORMAT(1X,/,' REPLICA',I5,' UU=',F10.3,' IMAX=',I5)                TTT04980
 910 FORMAT(1X,//,' LIEBERMAN ROSS SERIES,RL.8(',I4,')=',F12.5,' RL.9('TTT04990
    1,I4,')=',F12.5,//)                                                 TTTC5000
 911 FORMAT(1X,//,' XL8=',F12.5,' I5=',I5,' XL9=',F12.5,' I6=',I5)      TTTC5010
 999 FORMAT(1X,//' FOR COMPONENT ',I3,'THE NUMBER OF TRIALS IS ',I3)    TTT05020
1001 FORMAT(/,1X,'ACTUAL SYSTEM RELIABILITY IS ',F12.5)                 TTT05030
1000 FORMAT(1X,' CASE 12')                                              TTT05040
1002 FORMAT(/,1X,'       RL.8        RD.8        RT.8        RL.9        RDTTTC5050
    1.9         RT.9')                                                  TTTC5060
1003 FORMAT (1X,6F11.4)                                                 TTT05070
1004 FORMAT(1X,' AVERAGE FAILURES/REPLICA=',F10.3)                      TTTC5080
1005 FORMAT(/,1X,'  RL.8(400)  RD.8(400)  RT.8(400)  RL.9(450)  RD.9(45TTT05090
    1C)  RT.9(450)')                                                    TTT05100
1006 FORMAT(1X,' THE SAMPLE VARIANCES ARE ',/)                          TTTC5110
1008 FORMAT(1X,' THE SAMPLE MEANS ARE',/)                               TTTC5120
1007 FORMAT(1X,' BETA=',F5.2)                                           TTT05130
1009 FORMAT(1X,' FINAL=',F5.2,/)                                        TTTC5140
1010 FORMAT(1X,5F5.5)                                                   TTTC5150
1012 FORMAT (/,1X,' THE SAMPLE STANDARD DEVIATIONS ARE')                TTT05160
1014 FORMAT(1X,'FOR 1 REPLICA,MIN FAILURES=',I3,'   MAX=',I4)           TTT05170
1013 FORMAT(1X,' MCU TREL ')                                            TTTC5180
1020 FORMAT(1X,' NN(',I2,')=',I4)                                       TTTC5190
8C94 STOP                                                               TTTC5200
     DEBUG SUBCHK                                                       TTT05210
     END                                                                TTTC5220
     FUNCTION FCN(PNEWT)                                                TTT05230
     COMMON REALN1,UUU,ALPHA                                            TTTC5240
     PTEMP=FCN2(PNEWT,REALN1,UUU,ALPHA)                                 TTT05250
     FCN=PTEMP                                                          TTTC5260
     RETURN                                                             TTT05270
     END                                                                TTT05280
     FUNCTION FCN2(PNEWT,REALN1,UUU,ALPHA)                              TTTC5290
     N1=INT(REALN1)                                                     TTT05300
     TERM1=REALN1*PNEWT**(N1-1)*(1.-PNEWT)*(1.-UUU)                     TTTC5310
     TERM2=PNEWT**N1-ALPHA                                              TTTC5320
     FCN2=TERM1*TERM2                                                   TTT05330
     RETURN                                                             TTTC5340
     END                                                                TTT05350
```

LIST OF REFERENCES

1. Woods, W. M. and Borsting, J. R., A Method for Computing Lower Confidence Limits on System Reliability Using Component Failure Data with Unequal Sample Sizes, p. 1-23, U. S. Naval Postgraduate School, June 1968.

2. Gatliffe, T. R., Accuracy Analysis for a Lower Confidence Limit Procedure for System Reliability, Master Thesis, U. S. Naval Postgraduate School, September 1976.

3. Lockheed Missiles and Space Company, Inc., Publication LMSC-D368002A, Reliability Evaluation Plan for Trident I (C4) Missile, Appendix C, p. C-1 to C-34.

4. Lieberman, G. and Ross, S., "Confidence Intervals for Independent Exponential Series Systems," Journal of the American Statistical Association, p. 837-840, December 1971.

5. Fishman, G. S., Concepts and Methods in Discrete Event Digital Simulation, p. 168, Wiley, 1973.

6. U. S. Naval Postgraduate School Report NPS-53ZH76047 The Influence of Data Order on the Lieberman-Ross Method, by A. L. Schoenstadt, p. 1-37, April 1976.

7. Barr, D. R. and Jayachandran, R., "Improved Confidence Bounds Applied to Reliability," IEEE Transactions on Reliability, Vol. R-24, No. 1, p. 67-68, 1 April 1975.

8. Bazovsky, I., Reliability Theory and Practice, p. 238, Prenctice-Hall, 1961.

9. International Mathematical and Statistical Libraries, Inc., IMSL Library 1, FORTRAN IV, IBM S/370-360, p. MDCHI, VSORTA, ZREAL2, GGUB, IMSL, Houston, 1975.

10. Hutchinson, D. R., APL Histogram, Density Estimation, and Probability Plotting Routines, p. 10, 25, Master Thesis, U. S. Naval Postgraduate School, December 1976.

11. Subroutine Library of the W. R. Church Computer Center, U. S. Naval Postgraduate School, p. PLOTP, "Plot or Printer Multiple Curves," Monterey, 1974.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, California 93940	2
3.	Department Chairman, Code 55 Department of Operations Research Naval Postgraduate School Monterey, California 93940	1
4.	Dean W. M. Woods, Code 011 (thesis advisor) Dean of Educational Development Naval Postgraduate School Monterey, California 93940	1
5.	Professor R. R. Read, Code 55Re (second reader) Department of Operations Research Naval Postgraduate School Monterey, California 93940	1
6.	Professor A. L. Schoenstadt, Code 53Zh Department of Mathematics Naval Postgraduate School Monterey, California 93940	1
7.	Professor P. A. W. Lewis, Code 55Lw Department of Operations Research Naval Postgraduate School Monterey, California 93940	1
8.	LT. T. R. Maynard, USN VA-42, NAS Oceana Virginia Beach, Virginia 23460	1